

Procesi

- Kada se pokrene neki program (aplikacija, komanda) aktivira se jedan ili više procesa zavisno od složenosti samog programa
- Kada se program završi, procesi bi trebali u normalnoj situaciji da se automatski okončaju
- Program može biti interaktivan, gde korisnik ima kontrolu nad time šta program radi i kada će njegovo izvršenje da se okonča (npr. Vim editor), ali tu spadaju i programi koje korisnik samo pokrene poput ls komande i sličnih koje su rađene na dosadašnjim predavanjima
- Program može da bude i automatizovan (tzv. *batch* procesi) tj. ne mora biti vezan za terminal u kome korisnik zadaje komande tj. aktivira programe (na primer, to mogu biti zadaci/skripte koji su podešeni da se izvršavaju kad je opterećenje računara malo ili u zakazano vreme)
- Postoje i programi koji su konstantno aktivni (tzv. *daemons*), na primer, veb server koji mora neprestano da osluškuje da li je stigao zahtev za nekom stranicom

Procesi

- Sami procesi kada se pokrenu dobijaju svoj identifikacioni broj pomoću koga se identifikuju
- Svaki proces ima svog roditelja koji ga je kreirao (init proces je prvi proces koji se pokreće kada se učitava kernel)
- Kada proces završi sa svojim radom vraća tzv. izlazni kod svom roditelju koji potom treba da zatvori proces
- Može se desiti da proces mora duže da čeka da ga roditeljski proces zatvori (npr. programski kod nije dobro napisan, desi se bug) - tzv. zombi procesi (ne troše resurse sistema, ali zauzimaju id broj)
- Može se desiti da se roditeljski proces okonča pre završetka procesa potomka (tipično usled neke greške) - tzv. procesi siročad (tipično ih init proces uzme kao svoje direktne potomke)

Pregled procesa

- Postoji više komandi iz terminala koje daju uvid u trenutne procese, koliko resursa zauzimaju, njihove id-eve i sl.
- Naravno, postoje i grafičke aplikacije koje se mogu koristiti
- Komande **ps** i **top** daju prikaz liste procesa
- Razlika je što **ps** daje prikaz procesa za trenutak u kom je izdata komanda, a **top** vrši neprestano osvežavanje prikaza
- Otuda se **ps** komanda samo izvrši, ispiše listu procesa i potom se terminal oslobađa za unos naredne komande, a **top** komanda ostaje aktivna dok je korisnik sam ne terminira (slovo q vrši izlaz iz **top** aplikacije)
- Komanda **ps tree** daje prikaz stabla procesa i iz njega je lakše videti odnose između procesa i roditelj-potomak veze

ps

- Komanda koja daje pregled trenutnih procesa
- Navođenjem samo **ps** komande dobija se skraćeni prikaz procesa trenutnog korisnika i trenutnog terminala
- Na primer, ako bi korisnik otvorio dva terminala i u jednom od njih aktivirao **ps** komandu, ona bi prikazala samo procese vezane za dotični terminal (kao i proces koji predstavlja sam terminal), a ne i za onaj drugi terminal
- Skraćeni prikaz obuhvata id procesa, naziv komande koja je aktivirala proces, naziv terminala iz kog je aktiviran proces i CPU vreme koje je potrošeno na proces do tog momenta

ps

- Za detaljniji prikaz se koristi opcija -f
- Ona još dodatno prikazuje id korisnika koji je pokrenuo proces, id roditeljskog procesa, broj procesa potomaka, vreme starta procesa
- Opcija -l lista takođe detaljan prikaz - npr. prikazuje se i *nice* vrednost procesa, memorijski prostor koji zauzima proces (u veličini stranica - tipično jedna stranica je 4096B) i dr.
- Interesantno, -f i -l prikazi imaju međusobnih razlika tj. -l ne obuhvata sve što prikazuje -f, i obrnuto
- Za prikaz svih procesa se koristi opcija -e (ili -A koja je ekvivalentna)

ps

- Opcija -N je za negaciju selekcije
- Opcija -a selektuje sve procese osim lidera sesija i procesa koji nisu vezani (associated) za terminal
- Opcija -d selektuje sve procese sem lidera sesija
- Opcija -p *ID* daje prikaz za proces čiji je ID naveden (može se navesti i više ID-eva u liniji) - ekvivalenti su --pid i p
- Opcija --ppid *ID* daje prikaz procesa sa navedenim ID-em roditeljskog procesa
- Opcija -s *ID* daje prikaz procesa sa navedenim ID-em sesije
- Opcija -u *korisnik* daje prikaz procesa navedenog korisnika - ekvivalent je -U
- Opcija -C *komanda* daje procese koje je aktivirala navedena komanda

ps

- Opcijom `--format="lista"` se može definisati šta se želi prikazati u ispisu **ps** komande, gde lista sadrži kolone koje se žele prikazati (članovi liste se odvajaju space karakterom) - ekvivalent je opcija `-o`
- Opcijom `--sort="lista"` se može sortirati ispis **ps** komande, gde lista sadrži kolone za sortiranje (+ i - ispred člana liste određuje smer sortiranja)
- Postoje i brojne druge opcije, pri čemu se neke od njih ne navode sa - ispred svog naziva (razlika u odnosu na većinu komandi)

ps

- Oznake statusa:
 - D uninterruptible sleep (usually IO)
 - I Idle kernel thread
 - R running or runnable (on run queue)
 - S interruptible sleep (waiting for an event to complete)
 - T stopped by job control signal
 - t stopped by debugger during the tracing
 - W paging (not valid since the 2.6.xx kernel)
 - X dead (should never be seen)
 - Z defunct ("zombie") process, terminated but not reaped by its parent

ps

- Polja u prikazu procesa:
 - %cpu - prosečno iskorišćenje CPU
 - %mem - iskorišćenje memorije
 - args - komanda iza procesa
 - cmd - isto što i args
 - cputime - kumulativno CPU vreme u hh:mm:ss formatu
 - cputimes - kumulativno CPU vreme u sekundama
 - etime - proteklo vreme u hh:mm:ss formatu
 - etimes - proteklo vreme u sekundama
 - gid - group id
 - group - grupa

ps

- Polja u prikazu procesa:
 - ni - nice vrednost
 - pgid - proces ID grupe
 - ppid - ID roditeljskog procesa
 - pri - prioritet procesa
 - psr - kom procesoru je dodeljen proces trenutno
 - sess - ID sesije
 - sid - isto što i sess
 - start - vreme kad je komanda pokrenuta u hh:mm:ss
 - stat - status
 - tty - terminal
 - uid - ID korisnika
 - user - korisnik

Primer

```
ubuntu@ubuntu-VirtualBox:~$ ps
  PID TTY          TIME CMD
 8143 pts/0        00:00:00 bash
 8314 pts/0        00:00:00 ps
ubuntu@ubuntu-VirtualBox:~$ ps -f
UID          PID  PPID  C  STIME TTY          TIME CMD
ubuntu      8143  8133  0  14:17 pts/0        00:00:00 bash
ubuntu      8315  8143  0  15:31 pts/0        00:00:00 ps -f
ubuntu@ubuntu-VirtualBox:~$ ps -l
F S  UID      PID  PPID  C  PRI  NI ADDR  SZ  WCHAN  TTY          TIME CMD
0 S  1000    8143  8133  0  80   0  -   6734 wait  pts/0        00:00:00 bash
0 R  1000    8316  8143  0  80   0  -   3561 -     pts/0        00:00:00 ps
ubuntu@ubuntu-VirtualBox:~$ ps -l
F S  UID      PID  PPID  C  PRI  NI ADDR  SZ  WCHAN  TTY          TIME CMD
0 S  1000    8143  8133  0  80   0  -   6734 wait  pts/0        00:00:00 bash
0 R  1000    8317  8143  0  80   0  -   3561 -     pts/0        00:00:00 ps
ubuntu@ubuntu-VirtualBox:~$ ps -lf
F S UID          PID  PPID  C  PRI  NI ADDR  SZ  WCHAN  STIME TTY          TIME CMD
0 S ubuntu      8143  8133  0  80   0  -   6734 wait  14:17 pts/0        00:00:00 bash
0 R ubuntu      8318  8143  0  80   0  -   5669 -     15:32 pts/0        00:00:00 ps -lf
```

Primer

```
ubuntu@ubuntu-VirtualBox:~$ ps -ef
UID          PID    PPID  C  STIME TTY          TIME CMD
root         1      0   0  05:56 ?           00:00:01 /sbin/init splash
root         2      0   0  05:56 ?           00:00:00 [kthreadd]
root         3      2   0  05:56 ?           00:00:00 [ksoftirqd/0]
root         5      2   0  05:56 ?           00:00:00 [kworker/0:0H]
root         6      2   0  05:56 ?           00:00:02 [kworker/u2:0]
root         7      2   0  05:56 ?           00:00:00 [rcu_sched]
root         8      2   0  05:56 ?           00:00:00 [rcu_bh]
root         9      2   0  05:56 ?           00:00:00 [rcuos/0]
root        10      2   0  05:56 ?           00:00:00 [rcuob/0]
root        11      2   0  05:56 ?           00:00:00 [migration/0]
root        12      2   0  05:56 ?           00:00:00 [watchdog/0]
root        13      2   0  05:56 ?           00:00:00 [khelper]
root        14      2   0  05:56 ?           00:00:00 [kdevtmpfs]
root        15      2   0  05:56 ?           00:00:00 [netns]
root        16      2   0  05:56 ?           00:00:00 [perf]
root        17      2   0  05:56 ?           00:00:00 [khungtaskd]
```

Deo prikaza svih procesa. Proces init uvek dobija ID 1.

Primer

```
ubuntu@ubuntu-VirtualBox:~$ ps -p 8143
  PID TTY          TIME CMD
 8143 pts/0    00:00:00 bash
ubuntu@ubuntu-VirtualBox:~$ ps -lp 8143
F S  UID      PID  PPID  C  PRI  NI ADDR  SZ  WCHAN  TTY          TIME CMD
0 S  1000    8143  8133  0   80   0  -   6734 wait   pts/0    00:00:00 bash
ubuntu@ubuntu-VirtualBox:~$ ps -C bash
  PID TTY          TIME CMD
 8143 pts/0    00:00:00 bash
ubuntu@ubuntu-VirtualBox:~$ ps -lfc bash
F S  UID      PID  PPID  C  PRI  NI ADDR  SZ  WCHAN  STIME TTY          TIME CMD
0 S  ubuntu    8143  8133  0   80   0  -   6734 wait   14:17 pts/0    00:00:00 bash
ubuntu@ubuntu-VirtualBox:~$ ps -C bash --format="pid start cp sz"
  PID  STARTED  CP   SZ
 8143  14:17:58   0  6734
```

Primer

```
ubuntu@ubuntu-VirtualBox:~$ gnome-sudoku &
[1] 8839
ubuntu@ubuntu-VirtualBox:~$ ps -f
UID          PID    PPID  C  STIME TTY          TIME CMD
ubuntu      8829   8818   0  18:16 pts/0        00:00:00 bash
ubuntu      8839   8829   1  18:16 pts/0        00:00:00 gnome-sudoku
ubuntu      8842   8829   0  18:16 pts/0        00:00:00 ps -f
```

U terminalu je pokrenuta sudoku igra. Potom je terminal ugašen, ali je igra ostala aktivna i dalje. Na slici iznad se vidi da je u početku terminal bio roditeljski proces.

```
ubuntu@ubuntu-VirtualBox:~$ ps -f
UID          PID    PPID  C  STIME TTY          TIME CMD
ubuntu      8865   8854   0  18:17 pts/1        00:00:00 bash
ubuntu      8892   8865   0  18:18 pts/1        00:00:00 ps -f
ubuntu@ubuntu-VirtualBox:~$ ps -f -p 8839
UID          PID    PPID  C  STIME TTY          TIME CMD
ubuntu      8839   1092   0  18:16 ?            00:00:00 gnome-sudoku
ubuntu@ubuntu-VirtualBox:~$ ps -f -p 8839 1092
UID          PID    PPID  C  STIME TTY          STAT  TIME CMD
ubuntu      1092    904   0  06:36 ?            Ss    0:00 /sbin/upstart --user
ubuntu      8839   1092   0  18:16 ?            Sl    0:00 gnome-sudoku
```

U drugom terminalu koji je otvoren se ne vidi proces vezan za sudoku igru sa **ps -f** komandom jer nije otvorena iz tog terminala. Kada se ukuca prikaz procesa za sudoku igru vidi se da je sada ID roditeljskog procesa drugačiji od početnog.

Primer

```
ubuntu@ubuntu2004:~/fajlcount$ ps --format pid,ppid,pgid,sess,TTY,CMD
  PID   PPID   PGID   SESS TTY      CMD
  2055   2045   2055   2055 pts/0    bash
  3146   2055   3146   2055 pts/0    bash
  3318   3146   3318   2055 pts/0    ps --format pid,ppid,pgid,sess,TTY,CMD
ubuntu@ubuntu2004:~/fajlcount$ cat opc.sh | ps --format pid,ppid,pgid,sess,TTY,CMD
  PID   PPID   PGID   SESS TTY      CMD
  2055   2045   2055   2055 pts/0    bash
  3146   2055   3146   2055 pts/0    bash
  3323   3146   3322   2055 pts/0    ps --format pid,ppid,pgid,sess,TTY,CMD
```

Komanda može da pokrene jedan ili više procesa.

Proces pripada nekoj grupi procesa.

Grupa procesa pripada nekoj sesiji.

Primer: Pokreće se niz komandi u pajpu.

Svakoj komandi u pajpu odgovara jedan proces (pretpostavka je da je svaka komanda u pajpu pokrenula samo jedan proces).

Sam pajp predstavlja grupu procesa.

A terminal u kom je pokrenut pajp bi bio sesija kojoj pripada ta grupa procesa.

To ilustruje i slika na ovom slajdu. Objašnjenje slike je na sledećem slajdu.

```

ubuntu@ubuntu2004:~/fajlcount$ ps --format pid,ppid,pgid,sess,TTY,cmd
  PID   PPID   PGID   SESS TT      CMD
  2055   2045   2055   2055 pts/0   bash
  3146   2055   3146   2055 pts/0   bash
  3318   3146   3318   2055 pts/0   ps --format pid,ppid,pgid,sess,TTY,cmd
ubuntu@ubuntu2004:~/fajlcount$ cat opc.sh | ps --format pid,ppid,pgid,sess,TTY,cmd
  PID   PPID   PGID   SESS TT      CMD
  2055   2045   2055   2055 pts/0   bash
  3146   2055   3146   2055 pts/0   bash
  3323   3146   3322   2055 pts/0   ps --format pid,ppid,pgid,sess,TTY,cmd

```

Nalazimo se u bash potomku čiji PID (ID procesa) je 3146.

Prvi primer na slici prikazuje pokretanje ps komande. Ako se pogleda PGID (ID grupe procesa) vidi se da je on isti PID komande (tačnije procesa) ps. U ovom slučaju je ps komanda pripada grupi procesa koju sačinjava samo ps komanda. SESS prikazuje ID sesije, i on je jednak PID originalnog terminala (to je ID prvog procesa u sesiji, a to je originalno pokrenuti bash). Inače PPID ps komande je jednak PID bash potomka jer je u njemu aktivirana ps komanda.

Drugi primer je sličan samo je pokrenut pajp od dve komande. ID sesije je isti kao u prvom primeru što je očekivano jer je pajp pokrenut u okviru iste sesije kao prvi primer. PPID je isti jer je u istom bash potomku pokrenut pajp. Ali, PGID je sad drugačiji za ps komandu. Naime, sada grupu procesa sačinjavaju proces za cat komandu (njen proces nije ispisan u rezultatu ps komande jer je već završena) i proces za ps komandu. PGID se postavlja na PID prvog procesa koji je pokrenut u grupi procesa, a to je cat komanda u datom primeru. Otuda je PGID 3322. Iako cat nije prikazan u ispisu ps, znajući princip dodele vrednosti za PID, znamo da je proces za cat imao PID 3322.

top

- Komanda koja daje pregled trenutnih procesa u realnom vremenu
- Opcija *-d brojsekundi* omogućava definisanje perioda osvežavanja prikaza
- Opcija *-o kolona* omogućava sortiranje po navedenoj koloni (stavljanjem *+* ili *-* ispred naziva kolone se može definisati smer sortiranja)
- Opcija *-u korisnik* daje prikaz samo procesa vezanih za navedenog korisnika
- Dok je **top** aktivan, tasterom *f* se može ući u selekciju kolona za prikaz: tasterom *d* se može selektovati ili ukinuti kolona (toggle princip), tasterom *s* se može selektovati kolona za sortiranje, a tasterom *q* (ili Esc) se izlazi iz selekcije - ovo uputstvo se nalazi i na samom vrhu ekrana kada se uđe u deo za izbor kolona
- Postoje i druge opcije za dinamičko menjanje prikaza (npr. taster *s* za promenu vremena osvežavanja)

Primer

```
top - 15:47:35 up 9:51, 2 users, load average: 0.00, 0.04, 0.05
Tasks: 144 total, 1 running, 143 sleeping, 0 stopped, 0 zombie
%Cpu(s): 1.7 us, 0.0 sy, 0.0 ni, 98.3 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 1016868 total, 896176 used, 120692 free, 69292 buffers
KiB Swap: 1046524 total, 13132 used, 1033392 free. 244316 cached Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1263	ubuntu	20	0	1160696	150836	66600	S	1.0	14.8	32:52.49	compiz
1240	ubuntu	20	0	351156	6420	5512	S	0.3	0.6	0:30.27	ibus-daemon
8133	ubuntu	20	0	619016	31992	25348	S	0.3	3.1	0:06.88	gnome-terminal
1	root	20	0	117096	5160	3688	S	0.0	0.5	0:01.21	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.30	ksoftirqd/0
5	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/0:0H
6	root	20	0	0	0	0	S	0.0	0.0	0:02.35	kworker/u2:0
7	root	20	0	0	0	0	S	0.0	0.0	0:00.57	rcu_sched
8	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_bh
9	root	20	0	0	0	0	S	0.0	0.0	0:00.40	rcuos/0
10	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/0
11	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	migration/0
12	root	rt	0	0	0	0	S	0.0	0.0	0:00.46	watchdog/0
13	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	khelper
14	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kdevtmpfs

Pokrenuta **top** komanda bez opcija. Prikazan je deo procesa, a prikaz se periodično osvežava. Pritiskom na taster **q** se izlazi iz **top** aplikacije.

Primer

```
top - 15:52:57 up 9:56, 2 users, load average: 0.06, 0.09, 0.07
Tasks: 144 total, 1 running, 143 sleeping, 0 stopped, 0 zombie
%Cpu(s): 1.3 us, 0.3 sy, 0.0 ni, 98.3 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem: 1016868 total, 896444 used, 120424 free, 69324 buffers
KiB Swap: 1046524 total, 13132 used, 1033392 free. 244352 cached Mem
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
1	root	20	0	117096	5160	3688	S	0.0	0.5	0:01.21	systemd
2	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kthreadd
3	root	20	0	0	0	0	S	0.0	0.0	0:00.30	ksoftirqd/0
5	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	kworker/0:0H
6	root	20	0	0	0	0	S	0.0	0.0	0:02.35	kworker/u2:0
7	root	20	0	0	0	0	S	0.0	0.0	0:00.58	rcu_sched
8	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu_bh
9	root	20	0	0	0	0	S	0.0	0.0	0:00.40	rcuos/0
10	root	20	0	0	0	0	S	0.0	0.0	0:00.00	rcuob/0
11	root	rt	0	0	0	0	S	0.0	0.0	0:00.00	migration/0
12	root	rt	0	0	0	0	S	0.0	0.0	0:00.46	watchdog/0
13	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	khelper
14	root	20	0	0	0	0	S	0.0	0.0	0:00.00	kdevtmpfs
15	root	0	-20	0	0	0	S	0.0	0.0	0:00.00	netns

Pokrenuta **top** komanda sa opcijom -o -PID kojom se sortira prikaz po PID koloni. Obrnut redosled prikaza bi bio da je stavljeno +PID umesto -PID.

Primer

```
Fields Management for window 1:Def, whose current sort field is PID
Navigate with Up/Dn, Right selects for move then <Enter> or Left commits,
'd' or <Space> toggles display, 's' sets sort. Use 'q' or <Esc> to end!

* PID = Process Id          PGRP = Process Group I    vmj = Major Faults de
* USER = Effective User    TTY = Controlling Tty    vmn = Minor Faults de
* PR = Priority            TPGID = Tty Process Grp  USED = Res+Swap Size (
* NI = Nice Value         SID = Session Id        nsIPC = IPC namespace I
* VIRT = Virtual Image (  nTH = Number of Threa   nsMNT = MNT namespace I
* RES = Resident Size (   P = Last Used Cpu (     nsNET = NET namespace I
* SHR = Shared Memory (   TIME = CPU Time        nsPID = PID namespace I
* S = Process Status      SWAP = Swapped Size (K  nsUSER = USER namespace
* %CPU = CPU Usage        CODE = Code Size (KiB)  nsUTS = UTS namespace I
* %MEM = Memory Usage (R  DATA = Data+Stack (KiB
TIME+ = CPU Time, hundr  nMaj = Major Page Faul
* COMMAND = Command Name/Li
* PPID = Parent Process   nMin = Minor Page Faul
UID = Effective User      nDRT = Dirty Pages Cou
RUID = Real User Id      WCHAN = Sleeping in Fun
RUSER = Real User Name   Flags = Task Flags <sch
SUID = Saved User Id    CGROUPS = Control Groups
SUSER = Saved User Name  SUPGIDS = Supp Groups IDs
GID = Group Id          SUPGRPS = Supp Groups Nam
GROUP = Group Name      TGID = Thread Group Id
ENVIRON = Environment var
```

Prikaz ekrana nakon pritiska tastera *f* kojim se ulazi u selekciju kolona za prikaz i sortiranje. U man uputstvu **top** komande se mogu naći sva podešavanja koja su na raspolaganju.

top

- f taster za ulazak u kontrolni deo
- desna strelica za selekciju kolone, gore/dole strelice za promenu pozicije, i potom leva strelica ili enter za potvrdu
- d - toggle prikaza kolone
- s - izbor kolone za sortiranje (u delu prikaza sa R se menja smer sortiranja)
-

top

- U interaktivnom delu - space ili enter osvežavaju prikaz
- h ili ? daju prikaz pomoći
- r omogućava renice nekog procesa
- k omogućava gašenje procesa
- d omogućava podešavanje intervala osvežavanja
- R - menja smer sortiranja
- W - upis trenutnih podešavanja u konfiguracioni fajl

pstree

- Komanda koja daje prikaz stabla procesa
- Zgodno za vizuelnu analizu odnosa između procesa
- Ako se navede bez ID-a procesa prikazuje se kompletno stablo, u suprotnom se prikazuje podstablo čiji je koren navedeni proces
- Opcija -a prikazuje argumente komande koja je pokrenula proces ako je to moguće
- Opcija -p prikazuje ID procesa
- Opcija -n sortira po PID a ne po imenu procese istog roditelja
- Postoji i velik broj drugih opcija koje se mogu videti u uputstvu za **pstree** komandu

Primer

```
ubuntu@ubuntu-VirtualBox:~$ pstree
systemd--ModemManager--{gdbus}
                    --{gmain}
--NetworkManager--dhclient
                  --dnsmasq
                  --{NetworkManager}
                  --{gdbus}
                  --{gmain}
--accounts-daemon--{gdbus}
                  --{gmain}
--agetty
--avahi-daemon--avahi-daemon
--cgmanager
--colord--{gdbus}
         --{gmain}
--cron
--cups-browsed--{gdbus}
--cupsd--dbus
--dbus-daemon
--gnome-keyring-d--{gmain}
                  --{timer}
--kerneloops
--lightdm--Xorg
```

Prikazan je deo stabla.

Primer

```
ubuntu@ubuntu-VirtualBox:~$ pstree -ap
systemd,1 splash
├─ModemManager,495
│   ├──{gdbus},599
│   └──{gmain},592
├─NetworkManager,513 --no-daemon
│   ├──dhclient,7226 -d -q -sf /usr/lib/NetworkManager/nm-dhcp-helper -pf...
│   ├──dnsmasq,652 --no-resolv --keep-in-foreground --no-hosts ...
│   ├──{NetworkManager},636
│   ├──{gdbus},642
│   └──{gmain},641
├─accounts-daemon,494
│   ├──{gdbus},600
│   └──{gmain},594
├─agetty,843 --noclear tty1 linux
├─avahi-daemon,522
│   └─avahi-daemon,590
├─cgmanager,499 -m name=systemd
├─colord,1020
│   ├──{gdbus},1026
│   └──{gmain},1027
├─cron,521 -f
└─cups-browsed,604
```

Može se videti da su u stablu prikazani ID-evi procesa i opcije koje su korišćene za komande koje su pokrenule procese.

Primer

```
ubuntu@ubuntu-VirtualBox:~$ pstree -ap 513
NetworkManager,513 --no-daemon
├─dhclient,7226 -d -q -sf /usr/lib/NetworkManager/nm-dhcp-helper -pf...
├─dnsmasq,652 --no-resolv --keep-in-foreground --no-hosts --bind-interfaces--pi
├─{NetworkManager},636
├─{gdbus},642
└─{gmain},641
```

Zadavanjem ID procesa se može prikazati samo deo stabla tj. podstablo čiji je koren navedeni proces.

uptime

- Ova komanda prikazuje koliko dugo je sistem podignut

```
ubuntu@ubuntu-VirtualBox:~$ uptime
16:15:29 up 10:19, 2 users, load average: 0.01, 0.03, 0.05
ubuntu@ubuntu-VirtualBox:~$ uptime -p
up 10 hours, 19 minutes
```

Opcija -p ispisuje vreme u tzv. *pretty* formatu (tako je nazvano u uputstvu komande).

Terminiranje procesa

- Procesi bi trebali automatski da se terminiraju kad završe svoj posao (preciznije, kad proces završi posao, obaveštava svoj roditeljski proces koji ga potom terminira)
- Međutim, dešava se da je potrebno manuelno terminirati proces
- U tu svrhu se koristi komanda **kill**
- Ova komanda šalje zadati signal procesu koji treba da izazove okončanje procesa

kill

- Signal koji se šalje procesu **kill** komandom se zadaje ili nazivom signala ili brojem signala
- Lista naziva signala i njihovih brojeva se dobija sa **kill -l**
- Kada se želi terminirati proces zadaje se **kill** komanda koju prati broj ili naziv signala koji se šalje procesu i na kraju se navodi sam proces (njegov ID) - može se navesti i više procesa
- Signal SIGTERM (15) gasi proces regularno
- Signal SIGINT (2) daje instrukciju procesu da se ugasi, ali proces može ignorisati ovu instrukciju
- Signal SIGKILL (9) daje instrukciju procesu da se ugasi, i proces ne može ignorisati ovu instrukciju
- Signal SIGHUP (1) se koristi za daemon procese - daemon proces vrši ponovno čitanje svog konfiguracionog fajla

Primer

```
ubuntu@ubuntu-VirtualBox:~$ kill -l
 1) SIGHUP      2) SIGINT      3) SIGQUIT     4) SIGILL      5) SIGTRAP
 6) SIGABRT     7) SIGBUS     8) SIGFPE      9) SIGKILL     10) SIGUSR1
11) SIGSEGV    12) SIGUSR2   13) SIGPIPE    14) SIGALRM    15) SIGTERM
16) SIGSTKFLT  17) SIGCHLD   18) SIGCONT    19) SIGSTOP    20) SIGTSTP
21) SIGTTIN    22) SIGTTOU   23) SIGURG     24) SIGXCPU    25) SIGXFSZ
26) SIGVTALRM  27) SIGPROF   28) SIGWINCH   29) SIGIO      30) SIGPWR
31) SIGSYS     34) SIGRTMIN  35) SIGRTMIN+1 36) SIGRTMIN+2 37) SIGRTMIN+3
38) SIGRTMIN+4 39) SIGRTMIN+5 40) SIGRTMIN+6 41) SIGRTMIN+7 42) SIGRTMIN+8
43) SIGRTMIN+9 44) SIGRTMIN+10 45) SIGRTMIN+11 46) SIGRTMIN+12 47) SIGRTMIN+13
48) SIGRTMIN+14 49) SIGRTMIN+15 50) SIGRTMAX-14 51) SIGRTMAX-13 52) SIGRTMAX-12
53) SIGRTMAX-11 54) SIGRTMAX-10 55) SIGRTMAX-9 56) SIGRTMAX-8 57) SIGRTMAX-7
58) SIGRTMAX-6 59) SIGRTMAX-5 60) SIGRTMAX-4 61) SIGRTMAX-3 62) SIGRTMAX-2
63) SIGRTMAX-1 64) SIGRTMAX
```

Primer

```
ubuntu@ubuntu-VirtualBox:~$ gnome-sudoku &
[1] 8531
ubuntu@ubuntu-VirtualBox:~$ kill -15 8531
ubuntu@ubuntu-VirtualBox:~$ gnome-sudoku &
[2] 8534
[1] Terminated          gnome-sudoku
ubuntu@ubuntu-VirtualBox:~$ kill -SIGKILL 8534
ubuntu@ubuntu-VirtualBox:~$ gnome-sudoku &
[3] 8537
[2] Killed              gnome-sudoku
ubuntu@ubuntu-VirtualBox:~$ kill -9 8537
ubuntu@ubuntu-VirtualBox:~$
[3]+ Killed            gnome-sudoku
```

Kao što se vidi u datim primerima, može se navesti ime signala ili broj signala. Opcija -9 (tj. -SIGKILL) se tipično koristi kada se neka aplikacija zaglavi i regularno gašenje ne pomaže. U datim primerima su aplikacije pokretane u pozadini da bi terminal bio slobodan za izadavanje **kill** komande.

nice vrednost

- Tzv. *nice* vrednost procesa odgovara značaju tj. prioritetu procesa
- Što je veća vrednost to proces ima manji prioritet u pristupu procesoru
- Ova vrednost može biti i negativna (što je negativnija to proces ima veći prioritet)
- Kada neki proces zauzima velik procenat procesorskog vremena, može se pokušati promenom *nice* vrednosti tog procesa na manji prioritet
- Komanda **nice** *komanda* pokreće komandu sa difolt *nice* vrednošću koja je tipično 10 - kada se komanda pokreće bez **nice** onda je tipično *nice* vrednost 0
- Opcija *-n broj* definiše sa kojom *nice* vrednošću se pokreće komanda

Primer

```
ubuntu@ubuntu-VirtualBox:~$ nice ps -l
F S  UID  PID  PPID  C PRI  NI ADDR SZ  WCHAN  TTY          TIME CMD
0 S  1000  8521  8510  0  80   0  -  6735 wait  pts/0      00:00:00 bash
0 R  1000  8628  8521  0  90  10  -  3561 -      pts/0      00:00:00 ps
ubuntu@ubuntu-VirtualBox:~$ nice -n 5 ps -l
F S  UID  PID  PPID  C PRI  NI ADDR SZ  WCHAN  TTY          TIME CMD
0 S  1000  8521  8510  0  80   0  -  6735 wait  pts/0      00:00:00 bash
0 R  1000  8629  8521  0  85   5  -  3561 -      pts/0      00:00:00 ps
ubuntu@ubuntu-VirtualBox:~$ sudo nice -n '-5' ps -l
[sudo] password for ubuntu:
F S  UID  PID  PPID  C PRI  NI ADDR SZ  WCHAN  TTY          TIME CMD
4 S   0  8630  8521  0  80   0  -  14240 poll_s pts/0      00:00:00 sudo
4 R   0  8631  8630  0  75  -5  -  3561 -      pts/0      00:00:00 ps
ubuntu@ubuntu-VirtualBox:~$ sudo nice -n -5 ps -l
F S  UID  PID  PPID  C PRI  NI ADDR SZ  WCHAN  TTY          TIME CMD
4 S   0  8632  8521  0  80   0  -  14184 poll_s pts/0      00:00:00 sudo
4 R   0  8633  8632  0  75  -5  -  3561 -      pts/0      00:00:00 ps
```

Za upotrebu *nice* vrednosti koja je manja od difolt vrednosti kada se komanda pokreće regularno moraju se koristiti administratorske privilegije (sudo). Primititi da se -5 moglo navesti i bez navodnika jer *nice* iza opcije -n očekuje potencijalno i negativan broj. Namerno je u primerima rađeno pokretanje **ps -l** komande jer se odmah dobija listing *nice* vrednosti u NI koloni.

Promena nice vrednosti

- Nekada je potrebno promeniti *nice* vrednost već pokrenutog programa, na primer, kada neki proces zauzima velik procenat procesorskog vremena
- Komanda **renice** se koristi u tu svrhu
- Iza **renice** komande se prvo navodi nova *nice* vrednost, a potom se navodi redni broj procesa (može i više procesa da se navede)
- Ako se nova *nice* vrednost ne navodi odmah iza **renice** komande, onda se ispred nje mora navesti opcija -n
- Opcija -p *ID* definiše da se proces definiše preko ID-a procesa (ova opcija je difolt i ne mora striktno da se navede, sem ako ispred nje nisu korišćene druge opcije)
- Opcija -u *korisnik* navodi korisnika za čije procese se menja *nice* vrednost (može više korisnika da se navede)

Primer

```
ubuntu@ubuntu-VirtualBox:~$ gnome-sudoku &
[1] 8667
ubuntu@ubuntu-VirtualBox:~$ ps -l
F S    UID    PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
0 S    1000    8521 8510  0  80   0  -  6735 wait  pts/0      00:00:00 bash
0 S    1000    8667 8521  0  80   0  -  88208 poll_s pts/0      00:00:00 gnome-sudoku
0 R    1000    8670 8521  0  80   0  -  3561  -    pts/0      00:00:00 ps
ubuntu@ubuntu-VirtualBox:~$ renice 5 8667
8667 (process ID) old priority 0, new priority 5
ubuntu@ubuntu-VirtualBox:~$ ps -l
F S    UID    PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
0 S    1000    8521 8510  0  80   0  -  6735 wait  pts/0      00:00:00 bash
0 S    1000    8667 8521  0  85   5  -  88208 poll_s pts/0      00:00:00 gnome-sudoku
0 R    1000    8672 8521  0  80   0  -  3561  -    pts/0      00:00:00 ps
ubuntu@ubuntu-VirtualBox:~$ renice -n 10 -p 8667
8667 (process ID) old priority 5, new priority 10
ubuntu@ubuntu-VirtualBox:~$ ps -l
F S    UID    PID  PPID  C PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
0 S    1000    8521 8510  0  80   0  -  6735 wait  pts/0      00:00:00 bash
0 S    1000    8667 8521  0  90  10  -  88208 poll_s pts/0      00:00:00 gnome-sudoku
0 R    1000    8674 8521  0  80   0  -  3561  -    pts/0      00:00:00 ps
```

Primetiti da se opcije `-n` i `-p` ne moraju navoditi, ako se prvo navodi nova *nice* vrednost, a odmah potom i ID procesa za koga se menja *nice* vrednost (prvi primer). Drugi primer koristi `-n` i `-p` da se vidi da se te opcije uvek mogu napisati.

Pokretanje komande u pozadini

- Izvršenje neke komande (programa) može da potraje
- Ako takvu komandu pokrenemo u terminalu, terminal će biti zauzet sve dok komanda ne obavi svoj posao i korisnik neće moći unositi nove komande
- Pošto nije uvek poželjno da terminal bude 'zarobljen', postoji mogućnost pokretanja komande u pozadini
- Ako se na kraju komande navede & onda se komanda pokreće u pozadini (videli smo već primere pokretanja komandi u pozadini na ranijim slajdovima), a terminal ostaje slobodan za korisnika koji može da nastavi sa unosima novih komandi

Suspendovanje izvršenja komande

- Ako komanda nije pokrenuta u pozadini, a želimo da oslobodimo terminal, CTRL+Z suspenduje izvršenje komande i vraća terminal korisniku
- Suspendovana komanda se može ponovo pokrenuti i ona će nastaviti svoj rad od onog mesta do kog je stigla u trenutku svoje suspenzije
- Ako želimo da potpuno prekinemo rad komande i vratimo terminal korisniku može se iskoristiti CTRL+C
- Ovako okončana komanda je potpuno prekinuta i ako se ona želi ponovo izvršiti potrebno ju je ponovo pokrenuti

Pregled 'poslova'

- Komande koje su trenutno aktivne se nazivaju poslovi (jobs) - u suštini to su procesi komandi pokrenutih u terminalu
- Komandom **jobs** se dobija pregled trenutnih poslova
- Uz svaki posao stoji redni broj posla
- Uz pomoć rednog broja posao se može prebaciti u pozadinu (*background*) ili u terminal (*foreground* preciznije)
- Redni broj se može iskoristiti i samo za prikaz željenih poslova komandom **jobs** (ispred samog broja se navodi %)
- Opcija -l daje i ispis ID procesa
- Opcija -p prikazuje samo ID procesa
- Opcija -r prikazuje samo poslove koji se izvršavaju
- Opcija -s prikazuje poslove koji su zaustavljeni (suspendovani)
- Opcija -n prikazuje poslove koji su promenili status od zadnjeg ispisa

Prebacivanje komande u bg i fg

- Izvršenje komandi se može prebacivati iz bg (*background*) i fg (*foreground*)
- Komande **bg** i **fg** se koriste u tu svrhu, a njihov argument je redni broj posla
- Kada se navodi redni broj ispred samog broja se stavlja %
- Na taj način kad se sa CTRL+Z suspenduje neka komanda, ona može da nastavi sa svojim radom pokretanjem komande **fg** ili **bg** pri čemu se navodi redni broj suspendovang posla koji odgovara dotičnoj komandi
- Komanda **fg** će prebaciti izvršenje komande u terminal
- Komanda **bg** će prebaciti izvršenje komande u pozadinu

Primer

```
ubuntu@ubuntu-VirtualBox:~$ gnome-sudoku &
[1] 1806
ubuntu@ubuntu-VirtualBox:~$ gnome-mines &
[2] 1809
ubuntu@ubuntu-VirtualBox:~$ jobs
[1]-  Running                  gnome-sudoku &
[2]+  Running                  gnome-mines &
ubuntu@ubuntu-VirtualBox:~$ jobs -l
[1]-  1806 Running              gnome-sudoku &
[2]+  1809 Running              gnome-mines &
ubuntu@ubuntu-VirtualBox:~$ jobs -n
ubuntu@ubuntu-VirtualBox:~$ jobs -p
1806
1809
ubuntu@ubuntu-VirtualBox:~$ jobs %2
[2]+  Running                  gnome-mines &
ubuntu@ubuntu-VirtualBox:~$ jobs -l %2
[2]+  1809 Running              gnome-mines &
ubuntu@ubuntu-VirtualBox:~$ jobs -r
[1]-  Running                  gnome-sudoku &
[2]+  Running                  gnome-mines &
```


Primer

Sa **fg** komandom je izvršenje gnome-mines prebačeno u terminal čime je terminal zarobljen.

Sa CTRL+Z je suspendovan rad aplikacije gnome-mines čime je oslobođen terminal.

```
ubuntu@ubuntu-VirtualBox:~$ jobs -r
[1]-  Running                  gnome-sudoku &
[2]+  Running                  gnome-mines &
ubuntu@ubuntu-VirtualBox:~$ fg %2
gnome-mines
^Z
[2]+  Stopped                  gnome-mines
ubuntu@ubuntu-VirtualBox:~$ jobs -r
[1]-  Running                  gnome-sudoku &
ubuntu@ubuntu-VirtualBox:~$ jobs -s
[2]+  Stopped                  gnome-mines
ubuntu@ubuntu-VirtualBox:~$ bg %2
[2]+  gnome-mines &
ubuntu@ubuntu-VirtualBox:~$ jobs -r
[1]-  Running                  gnome-sudoku &
[2]+  Running                  gnome-mines &
ubuntu@ubuntu-VirtualBox:~$ jobs -s
```

Sa CTRL+C je prekinut rad aplikacije gnome-mines čime je oslobođen terminal. Može se videti da ova aplikacija više ne postoji u jobs jer je njen rad prekinut u potpunosti

```
ubuntu@ubuntu-VirtualBox:~$ jobs -r
[1]-  Running                  gnome-sudoku &
[2]+  Running                  gnome-mines &
ubuntu@ubuntu-VirtualBox:~$ fg %2
gnome-mines
^C
ubuntu@ubuntu-VirtualBox:~$ jobs
[1]+  Running                  gnome-sudoku &
```

Odloženo pokretanje komande

- Nekada je zgodno pokrenuti komandu (ili skriptu) odloženo, na primer, kada znamo da nećemo biti za računarom, a sama komanda (skripta) troši dosta sistemskih resursa
- Postoji više načina da se komanda pokrene u željenom trenutku
- Najjednostavniji način je **sleep vreme; komanda**
- Komanda **sleep** će obezbediti pauzu koja traje koliko je zadato *vreme*, i nakon toga će se pokrenuti izvršenje zadate komande (skripte)

```
ubuntu@ubuntu-VirtualBox:~$ sleep 5; find . -name *.pdf
./Documents/dir0/dir11/dir22/fff.pdf
./Documents/dir0/dir11/aa.txt.bb.pdf
./Documents/dir0/dir12/kf.pdf
./Downloads/intro-linux.pdf
```

U ovom primeru je zadato da se posle 5 sekundi pokrene **find** komanda.

Ako se želi izvršenje u pozadini, onda se ove dve komande mogu staviti u skriptu, a potom skriptu pokrenuti u pozadini. Druga varijanta je stavljanje komandi u zagradu iza koje se navodi & tj. (sleep 5; find . -name *.pdf)&

Komanda at

- Ova komanda se koristi za zadavanje trenutka izvršenja neke željene komande ili niza komandi
- Iza komande se navodi trenutak izvršenja, a potom se dobija prompt za unos željenih komandi
- CTRL+D vrši izlaz iz prompta i označava kraj unosa komandi koje želimo da se izvrše u zadatom trenutku
- Sam format vremena je veoma fleksibilan
- Midnight, noon, now, today, tomorrow, next week, 1430, 2:30 PM, 2:30 AM
- Može se operatorom + navesti i ofset u odnosu na neki zadati trenutak (minutes, hours, weeks, months, years - može da se koristi i jednina), takođe može da se koristi next umesto inkrementa za 1 npr. next hour je isto što i + 1 hour

Komanda at

- Opcija -l daje prikaz svih poslova trenutnog korisnika koji su zadati **at** komandom
- Opcija -f navodi fajl odakle treba da se pročitaju komande umesto da se unesu preko tastature
- Opcija -r omogućava da se neki posao obriše iz rasporeda (navodi se id tog posla)
- Komanda **atq** se takođe može iskoristiti za pregled svih poslova trenutnog korisnika zadatih **at** komandom
- Komanda **atrm** se može iskoristiti za brisanje nekog posla iz rasporeda (navodi se ID tog posla)


Primer

```
ubuntu@ubuntu-VirtualBox:~$ at now + 30 minutes
warning: commands will be executed using /bin/sh
at> echo Posao 1
at> <EOT>
job 6 at Sun Dec 11 13:12:00 2016
ubuntu@ubuntu-VirtualBox:~$ at 1730 next week
warning: commands will be executed using /bin/sh
at> echo Posao 2
at> <EOT>
job 7 at Sun Dec 18 17:30:00 2016
ubuntu@ubuntu-VirtualBox:~$ at 0800 + 2 days
warning: commands will be executed using /bin/sh
at> echo Posao 3
at> <EOT>
job 8 at Tue Dec 13 08:00:00 2016
ubuntu@ubuntu-VirtualBox:~$ atq
6          Sun Dec 11 13:12:00 2016 a ubuntu
8          Tue Dec 13 08:00:00 2016 a ubuntu
7          Sun Dec 18 17:30:00 2016 a ubuntu
ubuntu@ubuntu-VirtualBox:~$ atrm 6
ubuntu@ubuntu-VirtualBox:~$ atq
8          Tue Dec 13 08:00:00 2016 a ubuntu
7          Sun Dec 18 17:30:00 2016 a ubuntu
ubuntu@ubuntu-VirtualBox:~$ at -l
8          Tue Dec 13 08:00:00 2016 a ubuntu
7          Sun Dec 18 17:30:00 2016 a ubuntu
ubuntu@ubuntu-VirtualBox:~$ at -r 8
ubuntu@ubuntu-VirtualBox:~$ atq
7          Sun Dec 18 17:30:00 2016 a ubuntu
```

Primeri upotrebe at



Primer kada se komande nalaze u fajlu



```
ubuntu@ubuntu-VirtualBox:~$ at -f echo_skripta 0100
warning: commands will be executed using /bin/sh
job 9 at Mon Dec 12 01:00:00 2016
ubuntu@ubuntu-VirtualBox:~$ atq
9          Mon Dec 12 01:00:00 2016 a ubuntu
7          Sun Dec 18 17:30:00 2016 a ubuntu
```

Primer

```
ubuntu@ubuntu-VirtualBox:~$ atq
ubuntu@ubuntu-VirtualBox:~$ at now + 5 minutes
warning: commands will be executed using /bin/sh
at> echo EHO bez redirekcije
at> <EOT>
job 13 at Sun Dec 11 13:02:00 2016
ubuntu@ubuntu-VirtualBox:~$ at now + 5 minutes
warning: commands will be executed using /bin/sh
at> echo EHO sa redirekcijom >> ispis_aha
at> <EOT>
job 14 at Sun Dec 11 13:02:00 2016
ubuntu@ubuntu-VirtualBox:~$ atq
13      Sun Dec 11 13:02:00 2016 a ubuntu
14      Sun Dec 11 13:02:00 2016 a ubuntu
ubuntu@ubuntu-VirtualBox:~$ ls ispis_aha
ls: cannot access ispis_aha: No such file or directory
ubuntu@ubuntu-VirtualBox:~$ ls ispis_aha
ispis_aha
ubuntu@ubuntu-VirtualBox:~$ cat ispis_aha
EHO sa redirekcijom
ubuntu@ubuntu-VirtualBox:~$ atq
```

Primetiti da echo komanda koja nije redirektovana se nije pojavila u terminalu.

Komanda batch

- Nekada želimo da neku komandu (posao) izvršimo kada je sistem slabo opterećen
- Tada komanda **at** ne pomaže jer ona zahteva specificiranje egzaktnog vremena
- Stoga je kreirana komanda **batch** koja omogućava da se zadata komanda (komande) pokrene kad je opterećenje sistema ispod odgovarajuće granice (tipično 0.8)
- Komande se kao i kod **at** unose preko tastature (stdin) u otvorenom promptu (isto je CTRL+D za kraj unosa), a postoji i opcija -f kojom se može zadati fajl u kome se nalaze komande
- Za brisanje i pregled batch poslova se mogu koristiti iste komande kao i za at poslove

Cron i crontab

- Nekada želimo da neku komandu (posao) izvršavamo periodično, na primer svakog dana, svakog petka, svaka dva sata i sl.
- Komande **at** i **batch** se koriste da se željeni posao izvrši samo jednom
- Cron predstavlja daemon koji je neprestano aktivan i koji omogućava periodično izvršavanje poslova kad dođe trenutak za njihovo izvršenje
- Sami poslovi se navode u crontab fajlu (može biti više crontab fajlova)
- Komanda **crontab** se koristi za editovanje crontab fajlova, pregled njihovog sadržaja i sl.

crontab

- Opcija -e se koristi za editovanje crontab fajla
- Opcija -l se koristi za prikaz crontab fajla
- Opcija -r se koristi za brisanje crontab fajla
- Opcija -i radi isto što i -r, samo se dodatno postavlja pitanje korisniku da li je siguran da želi da obriše crontab fajl
- Opcija -u se koristi da se otvori crontab fajl navedenog korisnika

crontab

- Ako crontab fajl za korisnika ne postoji, pozivanje **crontab -e** će kreirati novi crontab fajl za korisnika
- Sam fajl na početku sadrži kratko uputstvo za pisanje poslova
- Svaki posao mora da se napiše u jednoj liniji koja sadrži 6 delova (delimiter je blanko karakter)
- Prvih 5 delova predstavljaju period, a šesti deo predstavlja posao (najbolje je posao staviti u skriptu i navesti apsolutnu putanju do skripte)
- Period se navodi u sledećem redosledu: minuta, sat, dan u mesecu, mesec, dan u nedelji
- Oznaka * označava sve

crontab

- Dani u nedelji se mogu navesti brojevima 0 do 7, gde 0 i 7 odgovaraju nedelji
- Dani u nedelji i meseci se mogu predstaviti i sa tri početna slova njihovog naziva (veličina karaktera nije bitna) - npr. Fri označava petak, Jan označava januar
- Opseg se navodi sa a-b, gde je a početna vrednost, a b krajnja vrednost opsega - npr. 1-5 u polju dani u nedelji bi predstavljao radne dane (ponedeljak - petak)
- Vrednosti se mogu odvajati i zarezima, npr., 7,14 u polju za sate bi predstavljao 7h i 14h
- Mogu se i kombinovati ova dva predstavljanja
- Ako se iza opsega navede / i vrednost onda navedena vrednost predstavlja korak u tom opsegu

Primer

```
ubuntu@ubuntu-VirtualBox:~$ crontab -e
no crontab for ubuntu - using an empty one

Select an editor. To change later, run 'select-editor'.
 1. /bin/ed
 2. /bin/nano      <---- easiest
 3. /usr/bin/vim.gnome
 4. /usr/bin/vim.tiny

Choose 1-4 [2]:
crontab: installing new crontab
```

Ako u direktorijumu /etc postoji cron.allow fajl korisnik mora biti naveden u njemu da bi mogao koristiti crontab. Ako u direktorijumu /etc postoji cron.deny fajl korisnik ne sme biti naveden u njemu da bi mogao koristiti crontab.

Primer

Ovde je prikazan sadržaj crontab fajla pozivom opcije -l. Ako se želi izbrisati neki unos ili dodati neki novi unos, onda se poziva crontab -e i potom se crontab fajl otvara u editoru i korisnik edituje sadržaj fajla.

```
ubuntu@ubuntu-VirtualBox:~$ crontab -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h dom mon dow   command
# PRIMER svaka dva sata petkom
0 */2 * * fri /home/ubuntu/echo_skripta
# PRIMER 15:30 svakog dana
30 15 * * * /home/ubuntu/echo_skripta
# PRIMER 08:00 i 14:00 svakog 15. u mesecu
0 8,14 15 * * /home/ubuntu/echo_skripta
```

```
ubuntu@ubuntu-VirtualBox:~$ crontab -r
ubuntu@ubuntu-VirtualBox:~$ crontab -l
no crontab for ubuntu
```

Opcija -r briše crontab fajl.