

Nadzor procesa i memorije

Par riječi o procesima

Proces u linuxu predstavlja objekt preko kojeg program koristi memoriju, procesorsko vrijeme i ulazno-izlazne resurse. U kernelu postoji tablica procesa - podatkovna struktura koja bilježi razne podatke o svakom pojedinačnom procesu (status procesa, adresni prostor, roditelj procesa, otvorene datoteke, ...). Ovdje ćemo navesti samo neke od parametara procesa koji imaju važnost za uobičajeno administriranje sustava.

- PID - Identifikator procesa (process ID) - Dinamički dodijeljen od kernela, po redu, kako je proces kreiran.
- PPID - Roditelj procesa (parent PID) - Svaki proces ima PPID veći od 1, osim u slučaju inita, "majke svih procesa" koji se pokreće prilikom inicijalizacije sustava i čiji je PPID 0.
- UID - Vlasnik procesa (user ID) - Brojčani identifikator vlasnika procesa tekstualno povezan s korisnikom preko datoteke /etc/passwd (broj u trećem stupcu).
- TTY - Upravljački terminal za koji je proces vezan.

Samo postojanje procesa ne mora značiti da isti automatski zauzima procesorsko vrijeme i ostale resurse. U osnovi moguća su četiri stanja procesa:

- Aktivni proces (runnable) - proces koji se izvršava
- Spavajući proces (sleeping) - proces koji očekuje specifični događaj. Primjerice demoni koji većinu svog vremena provedu u ovom stanju očekujući mrežnu konekciju.
- Zombi procesi (zombie) - Proces koji pokušavaju "umrijeti" - proces je završio ali je ostao zapis u tablici procesa.
- Zaustavljeni (stopped) - Proces koji je administrativno zaustavljen.

Nadzor procesa (ps)

Osnovni alat kojim se ispisuju procesi je ps. Pomoću te naredbe možemo dobiti sve podatke o trenutnim procesima ispisane u različitim formatima. Naredba ps postoji na svim UNIX-oidima, ali se njena implementacija znatno razlikuje od sustava do sustava. Na linuxu postoji objedinjena verzija koja se ponaša različito u ovisnosti o parametrima. Generalno možemo razlikovati standardnu i BSD sintaksu.

```
sh-3.1$ ps ajxf
PPID  PID  PGID  SID TTY      TPGID STAT  UID   TIME COMMAND
15222 15230 15222 15222 ?        -1 S     1000   0:00 sshd: ljubomir@pts/1
15230 15241 15241 15241 pts/1    15282 Ss     1000   0:00  \_ -bash
15241 15273 15273 15241 pts/1    15282 S      1000   0:00    \_ ksh
15273 15277 15277 15241 pts/1    15282 S      1000   0:00      \_ zsh
15277 15281 15281 15241 pts/1    15282 R      1000   0:00        \_ sh
15281 15282 15282 15241 pts/1    15282 R+     1000   0:00          \_ ps ajxf
```

1.1 Ispis ps naredbe u BSD stilu s parametrima za prikaz stabla procesa. Korisnik je više puta pokrenuo ljsku unutar ljske i na kraju izvršio naredbu ps ajxf. Obratite pažnju na PID i PPID identifikatore.

```
sh-3.1$ ps -ejH
  PID  PGID  SID  TTY          TIME CMD
15230 15222 15222 ?           00:00:00 sshd
15241 15241 15241 pts/1       00:00:00  bash
15310 15310 15241 pts/1       00:00:00  ksh
15323 15323 15241 pts/1       00:00:00  zsh
15326 15326 15241 pts/1       00:00:00  sh
15327 15327 15241 pts/1       00:00:00  ps
```

1.2 *Ista naredba pokrenuta u standardnom linux stilu (Zanimljivo je primjetiti razliku pri korištenju crtice ispred parametara kod ovog tipa sintakse).*

Kod uobičajenog linux kernela, bilo koji korisnik može dobiti izlist svih procesa na računalu. CARNet-ova standardna distribucija bazira se na Grsecurity dodacima kernelu kojima je običnom korisniku dozvoljen samo ispis vlastitih procesa, a za ispis svih procesa potrebno je koristiti root ovlasti.

Jedan od načina provjere procesa po potrošnji memorije je korištenje standardne naredbe ps aux uz dodatak parametara za sortiranje po količini memorije

```
korisnik@stroj:~$ sudo ps aux --sort -rss
USER      PID      %CPU    %MEM    VSZ      RSS
clamav    3205     0.1     11.9    133256   123480
amavis    17707    0.0     7.4     89084   77528
```

Stupac RSS (Resident Set Size) predstavlja dio memorije pojedinog procesa u RAM-u, a VSZ (Virtual memory SiZe) je veličina virtualne memorije procesa - obuhvaća RSS, memoriju u swapu te na file sistemu. Izražene vrijednosti su u kilobajtima.

- [Logirajte](#) [1] se za dodavanje komentara

pon, 2009-05-04 13:30 - Ljubomir Hrboka **Vote:** 4

Vaša ocjena: Nema Average: 4 (1 vote)

Naredba top (1)

Ukoliko želite provjeriti kako se u realnom vremenu troše resursi na vašem poslužitelju, praćenje procesa i memorije olakšat će vam naredba top. U sljedećih par članaka pregledat ćemo mogućnosti i ispis naredbe te objasniti njene pojedine djelove.

```
sudo top
```

```
top - 09:50:38 up 27 days, 18:09, 2 users, load average: 0.06, 0.03, 0.00
Tasks: 88 total, 2 running, 86 sleeping, 0 stopped, 0 zombie
Cpu(s): 20.3%us, 2.7%sy, 0.0%ni, 74.8%id, 2.3%wa, 0.0%hi, 0.0%si, 0.0%st
Mem: 1035524k total, 1020792k used, 14732k free, 68024k buffers
Swap: 979956k total, 3884k used, 976072k free, 403564k cached
```

PID	USER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COMMAND
16373	www-data	20	0	38108	19m	4284	S	21.3	2.0	0:00.80	apache2
30582	root	20	0	2252	1124	864	R	0.7	0.1	0:00.26	top
31949	mysql	20	0	124m	23m	4308	S	0.7	2.3	16:08.95	mysqld
2941	bind	20	0	38916	9500	1804	S	0.3	0.9	109:44.13	named

1. linija

```
top - 09:50:38 up 27 days, 18:09, 2 users, load average: 0.06, 0.03, 0.00
```

09:50:38 – trenutno vrijeme

up 27 days, 18:09 – poslužitelj je u pogonu već 27 dana 18 sati i 9 minuta

2 users – trenutni broj korisnika

load average: 0.06, 0.03, 0.00 – prosječan broj aktivnih procesa u zadnjih 1, 5 i 15 minuta

Ova linija je praktički ispis linux naredbe uptime. Prikaz ove linije u naredbi top može se isključiti/uključiti pritiskom na tipku l. Zanimljivo je obratiti pozornost na parametar "load average" koji nam govori koliko je procesa u navedena tri intervala koristilo ili tražilo korištenje CPU-a. Taj broj je prilično relevantan pokazatelj opterećenja poslužitelja. Kako bi analizirali njegovu vrijednost, broj je potrebno normalizirati s brojem procesora.

Primjerice, load average vrijednosti 1 na jednoprocorskom računalu je idealno opterećenje procesora. To bi značilo da je procesor u potpunosti opterećen, ali i da su svi procesi koji su zatražili njegovo korištenje to vrijeme i dobili. Ukoliko je broj veći od jedan, to znači da je broj procesa koji su zatražili korištenje procesora veći nego što mu dopuštaju vremenski intervali korištenja, odnosno procesor je zagušen zahtjevima. U našem slučaju je average load vrlo mali, gotovo blizu nule, što bi značilo da je ovaj poslužitelj u trenutku nastanka screenshota prilično neopterećen.

Na višeprocorskom računalu možemo zamisliti sljedeću situaciju: load average: 1.39, 2.62, 2.45. Ukoliko se, primjerice, radi o četveroprocorskom računalu možemo kazati da je prosječna potražnja za procesorskim vremenom u zadnjih 1 minutu bila otprilike 35%, u zadnjih 5 minuta otprilike 65%, a u zadnjih 15 minuta otprilike 61%, što bi značilo da su procesori prosječno opterećeni.

2. linija

```
Tasks: 88 total, 2 running, 86 sleeping, 0 stopped, 0 zombie
```

Prikazuje broj aktivnih, spavajućih, zombie i zaustavljenih procesa.

3. linija

```
Cpu(s): 20.3%us, 2.7%sy, 0.0%ni, 74.8%id, 2.3%wa, 0.0%hi, 0.0%si, 0.0%st
```

Prikazuje postotak vremena koje je procesor proveo izvršavajući aktivne procese. Ovo je razbijeno na postotke za pojedina područja:

user mod (us) – nesistemske (korisnički) programi – Ovdje bi trebao biti najveći dio vremena procesora. Tu su uključeni svi korisnički programi poput web, mail ili ftp servisa. Uobičajeno je da se ovaj parametar dinamički mijenja ovisno o količini korisnika na poslužitelju. Primjerice, ukoliko dosta posjetitelja istovremeno posjeti web stranice koje servisira naš poslužitelj, ovaj parametar će, u tom

trenutku, znatno porasti.

sistemski modu (sy) – sistemski programi (kernel) – Ovaj postotak bi se u normalno radu trebao zadržati na malim vrijednostima, pa njegovo povećanje može biti indikacija problema.

nice (ni) – vrijeme provedeno na programe čiji je prioritet izvršenja promijenjen naredbom nice

idle (id) – neiskorišteno procesorsko vrijeme – Veliki idle time ne znači ujedno i da je poslužitelj podopterećen. Ukoliko se na poslužitelju slabijih procesorskih mogućnosti dogodi prethodno navedena situacija kada, primjerice, 20-tak korisnika u istoj sekundi zatraže dinamički generiranu web stranicu, vrlo je velika vjerovatnost da će idle time u tom trenutku pasti na 0, a poslužitelj neće biti u stanju pravovremeno pružiti uslugu.

CPU IO vrijeme čekanja (wa) – postotak vremena koje procesor provede u čekajući na IO događaje (npr. čitanje s diska ili mreže). Velike vrijednosti mogu ukazivati da eventualni problem u radu ima veze s, primjerice, diskom.

hardwareski interrupti (hi) – vrijeme provedeno na posluživanje hardwareskih interrupta

softwareski interrupti (si) – vrijeme provedeno na posluživanje softwareskih interrupta

steal time (st) – vrijeme virtualnog procesora provedeno u nenamjernom čekanju dok hypervisor poslužuje drugi virtualni procesor

Druga i treća linija mogu se u radu isljučiti/uključiti pritiskom na tipku t.

4. i 5. linija

Ove linije su ekvivalent naredbi free. Četvrta linija prikazuje stanje memorije poslužitelja. U našem slučaju vidi se da poslužitelj ima 1GB ukupne memorije, Sljedeći stupac pokazuje da je zauzeto 1020792k, a treći da je 14732k slobodno. Ovdje treba napomenuti da je zauzeće memorije kompleksan podatak. U zauzetu memoriju linuxa treba ubrojiti, bolje rečeno, oduzeti posljednji stupac iz prikaza. Naime, stanje zauzete memorije ubraja i buffers prikazan u zadnjem stupcu četvrte linije te cached prikazan u zadnjem stupcu pete linije. Linux dinamički alokira buffer i cached memoriju po potrebi, primjerice pri svakom čitanju podataka s diska, a s nakanom da se što više podataka drži u cache-u kako bi im se omogućio brži pristup. U slučaju da neka korisnička aplikacija zatraži memoriju, linux smanjuje cache memoriju i dodjeljuje je aplikaciji. Dakle, podatak da naše računalo ima samo 14732k slobodno u stvari nije točan, jer se bilo kojem programu u svakom trenutku može osloboditi dio memorije u cacheu.

Možda čak bitniji podatak za kvalitetan rad računala je swap iz linije 5. Naime, računalo s dovoljnom količinom memorije ne bi nikad trebalo puniti swap memoriju na disku. Povećanje used swap memorije govori o tome da aplikacije zahtjevaju više memorije od postojeće, te da bi administrator trebao razmisliti o nadogradnji RAM-a.

Četvrta i peta linija mogu se u radu isljučiti/uključiti pritiskom na tipku m.

U sljedećem nastavku ćemo analizirati ispise pojedinih procesa, te način na koji taj ispis prilagoditi potrebama.

- [Logirajte](#) [1] se za dodavanje komentara

čet, 2009-05-28 10:27 - Ljubomir Hrboka **Vote:** 5

Vaša ocjena: Nema Average: 5 (1 vote)

Source URL: <https://sysportal.carnet.hr./node/572>

Links

[1] <https://sysportal.carnet.hr./sysportallogin>