

## Brisanje datoteka u direktoriju /tmp



Nedavno sam dobio upit od kolege sistemca mogu li se u svrhu povećanja diskovnog prostora, a bez previše razmišljanja, obrisati datoteke i direktoriji u /tmp. Odgovor bi bio u pravilu: ne!

Jedan od zadataka sistem-inženjera, odnosno administratora sustava je nadgledanje zauzeća diskovnog prostora, te naravno brisanje nepotrebnih datoteka iz privremenih direktorija, kao što je /tmp. No, ovdje treba biti oprezan jer se nepažljivim i nestručnim brisanjem može dovesti do nestabilnosti rada nekih aplikacija, pa i cijelog sustava (iako je ovo zapravo malo vjerojatno na modernim linuxima).

Većina operacijskih sustava pa tako i linux ima kreiranu particiju /tmp čija je svrha privremeno pohranjivanje datoteka. Razne aplikacije i programi koriste ovu particiju za spremanje privremenih datoteka koje su im potrebne u radu, FIFO datoteka itd. Primjerice, najpopularniji programi i servisi, kao što su apache, mysql, postgres koriste /tmp za pohranu privremenih datoteka.

Pogledajmo malo поближе kako izgleda direktorij /tmp i na koj način sve to funkcionira.

Pomoću naredbe **ls -ld** (opcija -d znači "ne prikazuj sadržaj direktorija, nego sam direktorij") možemo vidjeti dozvole /tmp direktorija:

```
# ls -ld /tmp/  
drwxrwxrwt 3 root root 16384 Feb 20 20:40 /tmp/
```

Na kraju prvog dijela ispisa, gdje se nalazi simbolički prikaz dozvola nad datotekom ili direktorijem, možemo vidjeti oznaku "t". Ovo nam govori da je postavljen **sticky bit**, što znači otprilike "samo vlasnik može brisati i mijenjati datoteke". Da sticky bit nije postavljen, svi korisnici bi mogli dodavati, mijenjati i brisati datoteke, neovisno o vlasniku (podrazumijevaju se dozvole 777, odnosno simbolički rwxrwxrwx). Root korisnik, naravno, može napraviti sve, bez obzira na *sticky bit*.

O brisanju datoteka koje je kreirao unutar direktorija /tmp, mora se brinuti i sam program, odnosno proces nakon što završi sa izvršavanjem. Inače, direktorij /tmp se briše kod svakog restarta poslužitelja, tako da ovdje nemojte spremati nikakve datoteke koje vam trebaju dulje vrijeme.

Međutim, ponekad se zna dogoditi (zbog loše napisanog programa ili neke druge greške) da program nakon završetka izvršavanja ne obriše privremene datoteke i s vremenom se direktorij /tmp popuni. Ovime sami sebi možemo napraviti DoS (Denial of Service) napad, što može prouzrokovati da ostali programi više ne mogu raditi.

Ako se dogodi da je /tmp zapunjen, moramo na kontroliran i regularan način obrisati privremene datoteke:

```
# df -ha  
Filesystem Size Used Avail Use% Mounted on  
:  
/dev/sda6 1.5G 1.5G 0G 100% /tmp  
/dev/sda7 182G 19G 164G 11% /home
```

Pokušat ćemo naći koja datoteka ili datoteke su zauzelo toliko mjesta na inače praznoj particiji gdje nam se nalazi /tmp. Pokušajmo obični ls:

```
/tmp# ls -la
drwxrwxrwt  3 root    root    16K Mar 11 13:26 .
drwxr-xr-x 24 root    root    1.0K Dec 17 12:40 ..
drwx-----  2 root    root    16K Aug 16  2005 lost+found
-rw-----  1 postfix munin    0 Mar 10 11:35 sh-thd-4081321527
```

ili naredbu **du** -ha:

```
/tmp# du -ha
16K    ./lost+found
0      ./sh-thd-4082268886
0      ./sh-thd-4082269878
```

Iz primjera vidimo da je direktorij /tmp u potpunosti zauzet (100%), no iz nekog čudnog razloga ne vidimo datoteke koji su zauzele toliko prostor. Naravno, ono što ne vidimo ne možemo obrisati. Što je u tom trenutku najbolje napraviti? Na sreću, imamo još alata koje možemo upotrijebiti.

Iskoristit ćemo naredbu **lsdf**, koja nam omogućuje da vidimo sve otvorene datoteke na sustavu, uključujući i mrežne priključke (to nam u ovom trenutku ne treba). Ispis naredbe ćemo malo suziti, odnosno grepati, a na taj način možemo vidjeti koliko je neki proces sa svojim privremenim datotekama zauzeo prostora u direktoriju /tmp:

```
# lsdf | grep /tmp
mysqld    2120      mysql    7u      REG      8,8      1610612736 15 /tmp/ib6Two
I9 (deleted)
apache2   2369      root    DEL      REG      8,8      17 /tmp/apc.Zk9
WUt
apache2   2377      www-
data DEL      REG      8,8      17 /tmp/apc.Zk9WUt
apache2   2378      www-
data DEL      REG      8,8      17 /tmp/apc.Zk9WUt
:
:
apache2   2397      www-
data DEL      REG      8,8      17 /tmp/apc.Zk9WUt
apache2   2398      www-
data DEL      REG      8,8      17 /tmp/apc.Zk9WUt
:
amavisd-n 32140     amavis   13u     REG      8,3      26601      892767 /var/lib/ama
vis/tmp/amavis-20130311T125221-32140/email.txt
```

U izlistu vidimo krivca, a to je proces *mysqld* koji je sa svojom datotekom zauzeo 1.6 GB. Također možemo vidjeti i da je ta datoteka obrisana "(deleted)". Pa zašto onda još zauzima prostor?

Generalno gledajući, na Unixima, a i Linuxu, kad proces otvori datoteku stvori se pokazivač *file descriptor* (FD), koji može preživjeti brisanje datoteke. Proces i dalje može pisati u tu datoteku i čitati iz nje, kao da i dalje postoji. Drugim procesima i sustavu ta datoteka nije vidljiva (ls, stat i drugi alati je ne mogu vidjeti), ali prostor kojeg opisuje taj FD se ne oslobađa. Do ovoga dolazi obično nakon nadogradnji sustava, ali može se pojaviti i u drugim slučajevima. Rješenje je reboot sustava.

U slučaju da ne želimo raditi reboot sustava, (u ovom primjeru) dovoljno je zaustaviti proces *mysqld*, pričekati par sekundi i ponovo ga startati:

```
# /etc/init.d/mysqld stop
# lsof | grep /tmp
bash          437      root  cwd      DIR      8,6     16384      2 /tmp
apache2       1837     root  DEL      REG      8,6           12 /tmp/apc.T3o
LG0
mysqld        2156     mysql 12u     REG      8,6           0       17 /tmp/iba0E1N
G (deleted)
amavisd-
n 4508      amavis  cwd      DIR      8,3     4096     892328 /var/lib/amavis/tmp
```

Zaustavljanjem procesa mysqld i njegovim ponovnim pokretanjem prostor je uspješno oslobođen:

```
# df -ha
Filesystem          Size  Used Avail Use% Mounted on
/dev/sda6            1.5G   54M  1.4G   4% /tmp
```

i particija je ponovo slobodna.

Procedura bi ukratko bila:

1. lsof | grep /tmp
2. zaustavljanje procesa koji je zauzeo particiju
3. pokretanje tog procesa

Ako gornji recept ne pomaže, preostaje nam napraviti *reboot* sustava i tako počistiti /tmp direktorij.

Zdravko Rašić

sri, 2013-03-20 07:01 - Zdravko Rašić **Kuharice:** [Linux](#) [1]

**Kategorije:** [Software](#) [2]

**Vote:** 0

No votes yet

**Source URL:** <https://sysportal.carnet.hr/node/1229>

### Links

[1] <https://sysportal.carnet.hr/taxonomy/term/17>

[2] <https://sysportal.carnet.hr/taxonomy/term/25>