Sprječavanje fork-bombe

Kao prvo, što je fork-bomba? Fork-bomba je oblik napada ukraćivanjem reurrsa (Denial of Service), kada se jedan proces kontinuirano umnožava kako bi usporio rad ili čak srušio računalo na kojem se izvršava. S druge strane, nedostatak reusursa može spriječiti normalan rad "legalnih" servisa. Prisjetite se članka Apache: poruka "resource temporarily unavailable [1]", koji je ukratko odgovorio na problem nestabilnog rada poslužitelja Apache. Radilo se o broju dopuštenih otvorenih datoteka, a problem se rješavao pomoću naredbe "ulimit". Upravo tu naredbu ćemo opisati u nastavku.

Ukratko, naredbom ulimit možete spriječiti "fork-bombu". Fork-bombu moguće je pokrenuti iz naredbene linije u sljedećem obliku (ovo nemojte raditi na produkcijskom poslužitelju, iz razumljivih razloga):

```
korisnik@debian:~$      :(){ :|:& };:
```

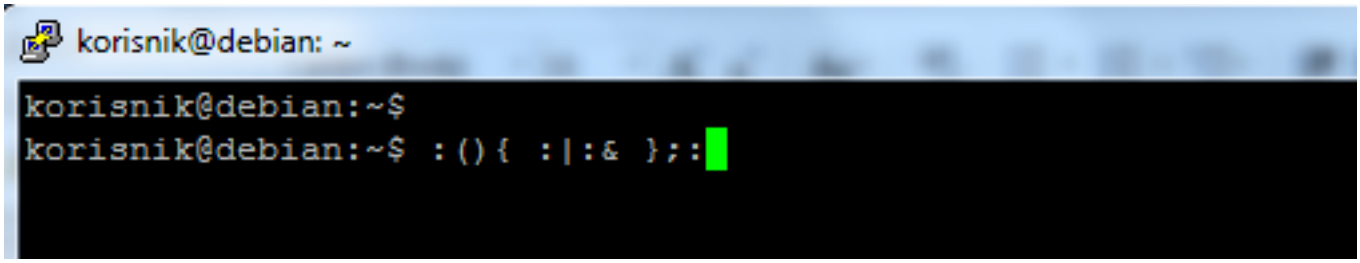
Ukratko ćemo objasniti što je što unutar ovog *onelinera*:

:() - definira funkciju(bez argumenata) ":"
{:|:&} - pokreni funkciju ":", proslijedi rezultat ":" ponovo funkciji i postavi u pozadinu
; - kraj skripte kako bi mogla započeti
: - pokreni funkciju, lančana reakcija

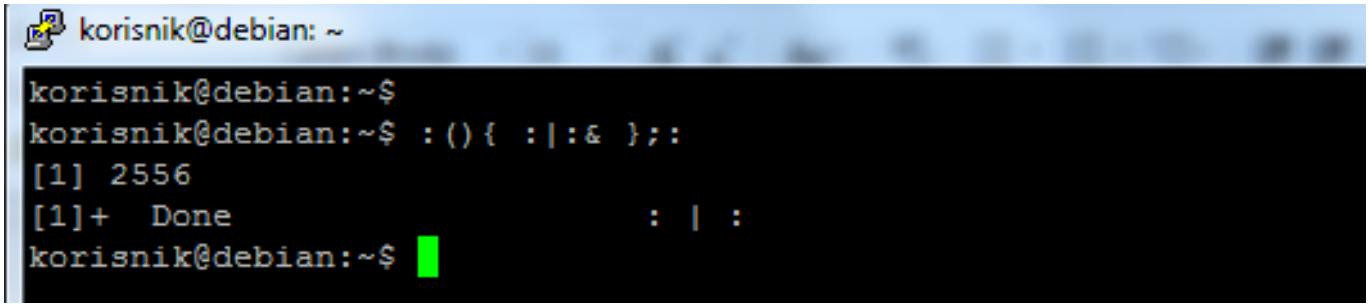
Problem na kojeg nailazimo je taj što fork-bombu može pokrenuti običan korisnik prijavljen u linuxovu ljusku (*shell*). Ukoliko to na sustavu nije administrativno zabranjeno, korisnik može povećati "mekanu" granicu (soft limit), odnosno broj dopuštenih otvorenih datoteka. Ako granica uopće ne postoji (hard limit), to znači da je reusurse moguće iscrpiti do kraja.

Prvi primjer koji ćemo prikazati je izvršen na svježe instaliranom debianu, sa standardnim paketima. Na sustavu je ostavljen *default*, što znači da nije mijenjan broj dopuštenih otvorenih datoteka.

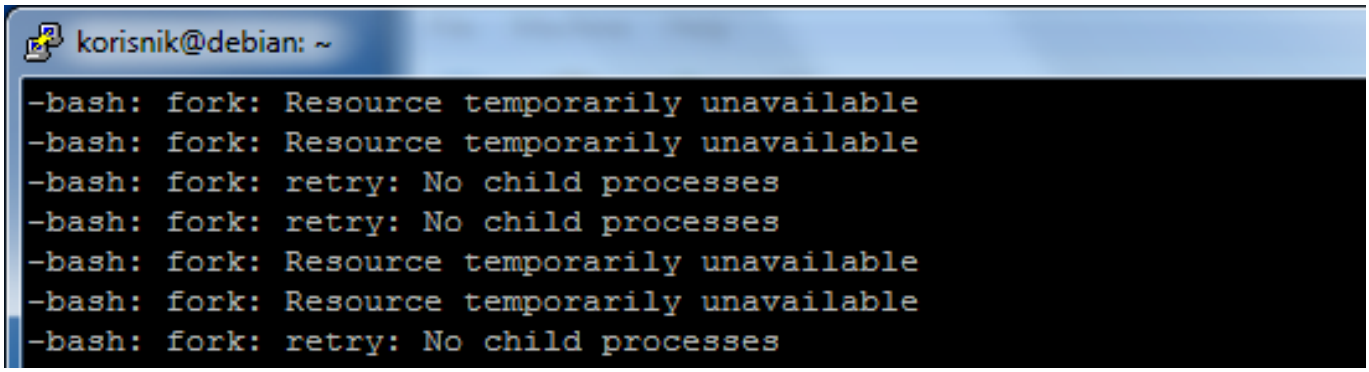
Korisnik "korisnik" ukucao je bash skriptu :(){ :|:& };:



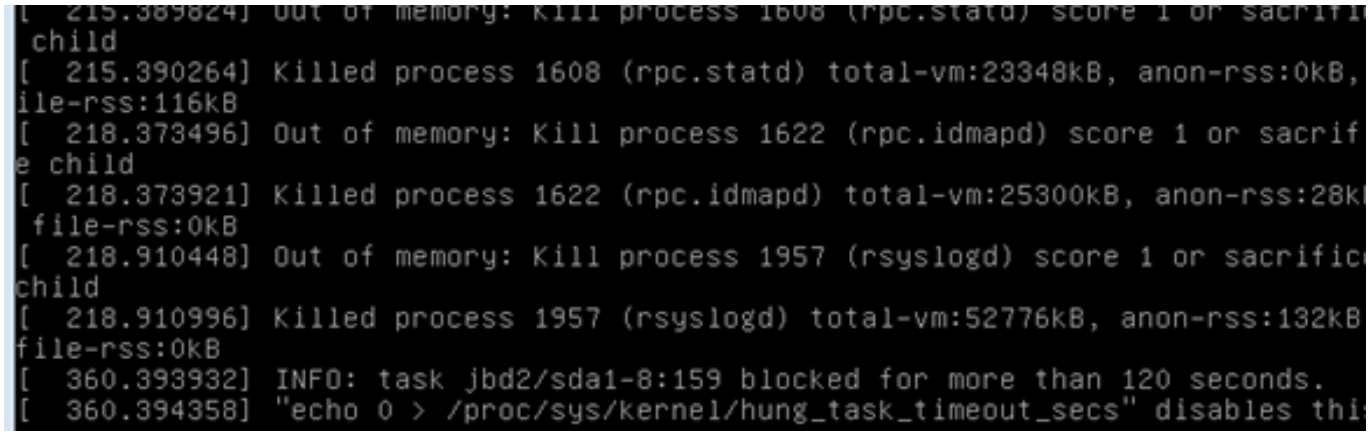
Nakon pokretanja, sustav više nije bio dostupan, nije bila moguća prijava čak ni root korisnika, a skripta se nije mogla zaustaviti s CTRL+C ili bilo kojom drugom kombinacijom tipki:



Prilikom izvršavanje skripte korisnik koji ju je pokrenuo vidi "uspješan" rezultat:



Pokušaj prijave kao root s konzole ostaje samo pokušaj, a na konzoli vidimo hrpu procesa:



Jedino što Vam preostaje je isključiti poslužitelj, ali samo pomoću tipke "Power off" na kućištu računala i ako ste imali sreće da je to bilo za vrijeme radnog vremena. No, ako se taj događaj dogodio izvan radnog vremena, nema vam druge nego nazad na svoje radno mjesto. "Na daljino" ništa ne možete napraviti.

Kako bi izbjegli ove situacije možete onemogućiti terminalski pristup korisnicam (ostaviti pristup administratoru sistema). Ukoliko to u vašem slučaju nije moguće (jer neki korisnici trebaju takav pristup), možete probati ograničiti broj otvorenih datoteka i procesa.

Prvo provjerimo trenutno stanje otvorenih datoteka s naredbom ulimit (ono što vidimo je da je *po defaultu* maksimalan broj dopuštenih procesa postavljen na 27730, a broj dopuštenih istovremeno otvorenih datoteka na 1024):

```
root@debian:~# ulimit -a
core file size(blocks, -c) 0
data seg size (kbytes, -d) unlimited
```

```
scheduling priority (-e) 0
file size (blocks, -f) unlimited
pending signals (-i) 27730
max locked memory (kbytes, -l) 64
max memory size (kbytes, -m) unlimited
open files (-n) 1024
pipe size (512 bytes, -p) 8
POSIX message queues (bytes, -q) 819200
real-time priority (-r) 0
stack size (kbytes, -s) 8192
cpu time (seconds, -t) unlimited
max user processes (-u) 27730
virtual memory (kbytes, -v) unlimited
file locks (-x) unlimited
```

Kreirajmo novu datoteku **/etc/security/limits.d/00-stopfork.conf**. U njoj ćemo ograničiti broj istovremeno otvorenih datoteka na 300 i procesa na 300 (postavljamo i soft i hard ograničenje):

```
#
*       soft    nproc   300
*       hard    nproc   500
*       soft    nofile  300
*       hard    nofile  500
```

Iz gornjeg primjera možete primjetiti u kojem obliku se postavlja unos u datoteku 00.nesto.conf.

U stupac **<domain>** unosimo korisničko ime, ime grupe (gdje za ime grupe koristimo oblik @nazivgrupe), ali možemo koristiti zamjenske znakove "*****" i "**%**".

Zamjenski znak "*" odnosi se na sve korisnike, dok "%" koristimo kad želimo ograničiti maksimalan broj prijava (logiranja) za korisnika.

Stupac **<type>** sadrži vrstu ograničenja (limit) "**soft**" i "**hard**" i "**-**" (minus).

Soft ograničenje omogućuje običnom korisniku da proizvoljno povećava ili smanjuje ograničenja sve do hard ograničenja.

Hard ograničenje postavlja administrator sustava koje običan korisnik ne može zaobići, tj. prijeći maksimalno postavljenu vrijednost.

Minus ("**-**") (bez navodnika) označava da zapravo forsirate izjednačavanje soft i hard ograničenja.

Stupac **<item>** sadrži naziv stavki za koje se postavlja ograničenje (core, nofile, cpu, nproc, maxlogins, nice...) i na kraju imamo stupac **<value>,** u kojeg upisujemo brojčane vrijednosti ograničenja za pojedine stavke.

Primjer kreiranja ograničenja za grupu student:

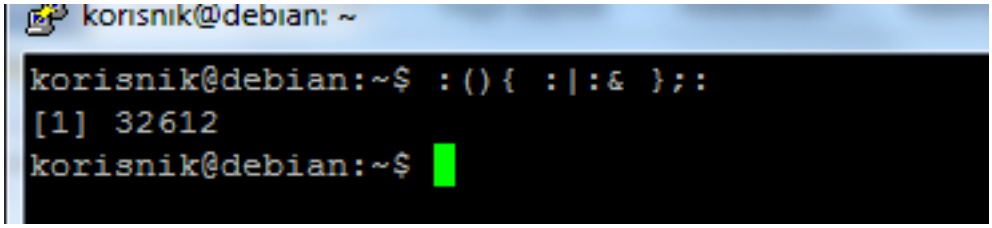
```
<domain>        <type>              <item>          <value>
@student        hard                nproc           200
@student        -                   maxlogins       4
```

Nakon unosa, dovoljno je napustiti terminal i ponovo se prijaviti. Nakon toga možete provjeriti da li su izmjene prihvaćene:

```
debian:~$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
scheduling priority             (-e) 0
file size               (blocks, -f) unlimit
pending signals                 (-i) 27730
max locked memory       (kbytes, -l) 64
max memory size         (kbytes, -m) unlimited
open files                      (-n) 300
pipe size            (512 bytes, -p) 8
POSIX message queues     (bytes, -q) 819200
real-time priority              (-r) 0
stack size              (kbytes, -s) 8192
cpu time               (seconds, -t) unlimited
max user processes              (-u) 300
```

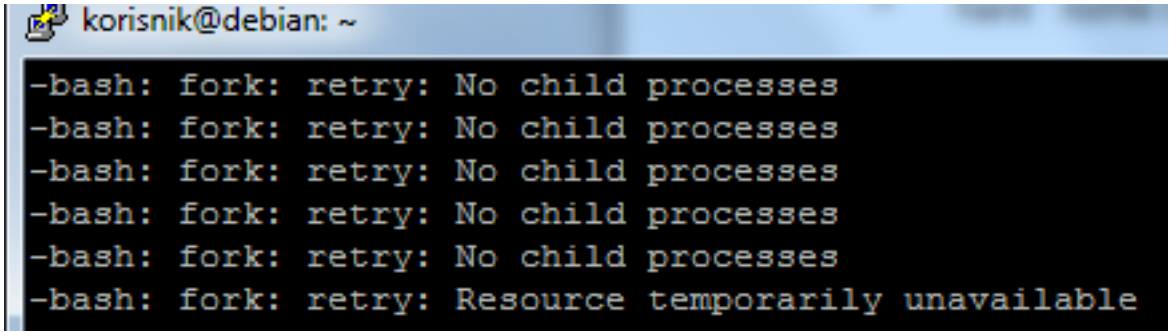
Vidimo da je prihvaćen soft limit na 300, te sada obični korisnik može mijenjati broj otvorenih datoteka do ove vrijednosti. Iako se u ispisu to ne vidi, korisnik ne može prijeći broj definiran u hard ograničenju od 500 procesa, postavljenog od strane administratora.

Pogledajmo kako će se sad ponašati poslužitelj:

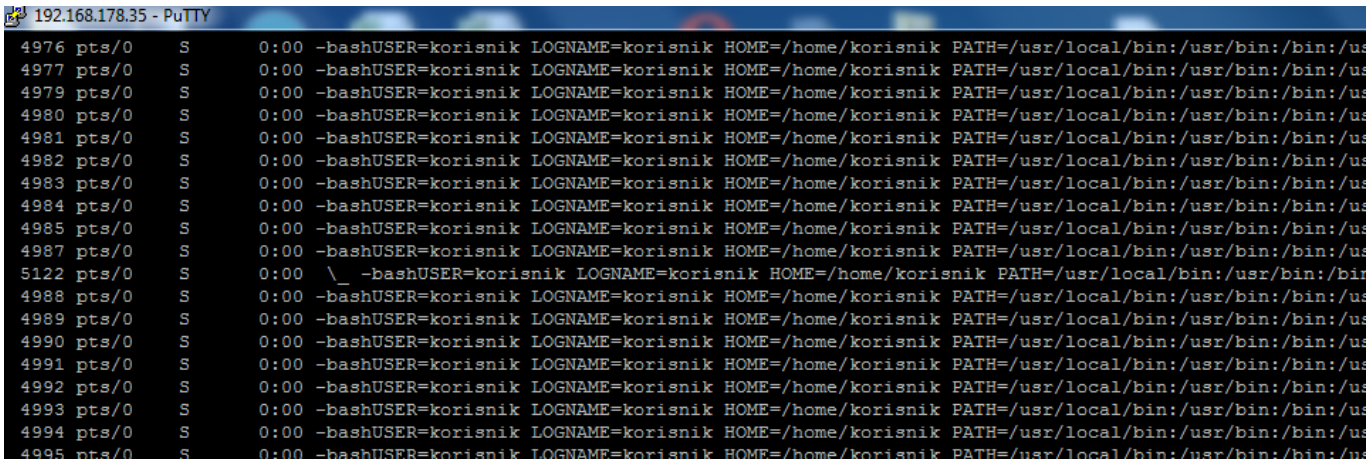


Nakon pokretanja skripte korisnik dobije sličan ispis kao i u prvom slučaju, ali sada skriptu može zaustaviti običnim CTRL+C, pristup administratoru nije onemogućen, te se može udaljeno napraviti zaustavljanje skripti ili napraviti dodatne izmjene.

Administrator se normalno prijavljuje:



Sada imamo mogućnost vidjeti procese s naredbom "ps". Ako malo pogledamo procse s "ps -aex", možemo vidjeti tko se "igra" na poslužitelju:



To je upravo naš standardni i dežurni krivac, "korisnik".

uto, 2015-02-24 20:47 - Zdravko Rašić**Vijesti:** Linux [2]
Kuharice: Linux [3]
Vote: 5

Vaša ocjena: Nema Average: 5 (2 votes)

Source URL: https://sysportal.carnet.hr/node/1522

Links
[1] https://sysportal.carnet.hr/node/1187
[2] https://sysportal.carnet.hr/taxonomy/term/11
[3] https://sysportal.carnet.hr/taxonomy/term/17