# Filtriranje mrežnog prometa na aplikacijskoj razini pomoću Linux rješenja - 1.dio

**Uvod**
Mrežni filtar tj. firewall postao je dio "mrežne svakodnevice" sredinom 80-tih godina prošlog stoljeća, još od pojave usmjernika (routera). Velika većina firewalla ima mogućnost filtriranja mrežnog prometa isključivo na drugom, trećem i četvrtom OSI sloju (podatkovni, mrežni i prijenosni slojevi), tj. prema MAC adresi, IP adresi, priključnoj točki (portu usluge) i stanju konekcije. Sve masovnija upotreba računalnih mreža dovodi do potrebe kontroliranja mrežnog prometa na sedmom sloju tj. aplikacijskoj razini.

Mrežni se promet filtrira prema protokolima koji se koriste na ovoj razini OSI modela, pa čak i prema samim vrstama datoteka koje se prenose mrežom. Uz komercijalne proizvode (npr. Check Point VPN-1) pojavila su se i rješenja bazirana na open-source tehnologijama, a jedno takvo rješenje objašnjeno je u ovom članku.

Kako ovaj postupak zahtijeva "patchiranje" i (pre)kompajliranje kernela, preporuka je da ovo ne pokušavate na produkcijskom poslužitelju, nego da za ovu namjenu izdvojite zasebno računalo, koje ne mora biti najnovije. Tako ulogu mrežnog filtra i DHCP poslužitelja za otprilike 50-ak računala, u mom slučaju, sasvim zadovoljavajuće obavlja Pentium III na 800MHz s 256MB radne memorije, tvrdim diskom od 20GB i dvije mrežne kartice.

Za početak, potrebno je ukratko objasniti nama interesantan način rada mrežne kartice. Red čekanja (*queue*) je privremena memorija (*buffer*) ili lokacija, koja sadrži konačan broj paketa, koji čekaju da se nad njima obavi neka akcija. Svako mrežno sučelje ima svoj predefinirani red čekanja, a u njemu postoje 3 moguće akcije: ulazak paketa u red (*enqueue*), njegov izlazak iz reda (*dequeue*) i njegovo brisanje (*drop*). U najjednostavnijem se obliku paketi u redu čekanja prosljeđuju po FIFO principu i bez drugih mehanizama nije moguće kontrolirati njihovo ponašanje. Metode upravljanja redovima (*queuing disciplines - qdiscs*) su algoritmi kojima se kontrolira način ulaska paketa u redove i njihovog izlaska. Ukratko, potrebno je prepoznati željeni protokol tj. paket koji pripada interesantnom toku podataka (*match*), nekako ga označiti (*mark*) i onda ga na neki način oblikovati (*shape*). Više o tome možete pročitati u Linux Advanced Routing and Traffic Control HOWTO dokumentu (lartc.org [1]).

**Instalacija**
Sama instalacija nije komplicirana, a najviše vremena oduzima (pre)kompajliranje kernela.
Potrebni paketi na računalu su:
- 2.4 ili 2.6 kernel source (kernel.org [2]) - preferirano 2.6
- iptables [3] source
- iproute2 [4]
- l7-filter [5]
- definicije protokola [6]
Napomena: trenutna verzija l7 filtra nije kompatibilna s kernel verzijom 2.6.20.x.

Za potrebe ovog članka svi paketi skinuti su u direktorij /usr/src/.

Da bi uključili podršku za l7 filtar, potrebno je patchirati kernel source i iptables. Patchevi za odgovarajuće verzije nalaze se unutar l7-filter paketa, pa prvo raspakiramo paket l7-filter u proizvoljni direktorij (u primjeru je korišten direktorij /usr/src/netfilter-layer7):

```
tar xzvf netfilter-layer7-vX.Y.tar.gz /root/netfilter-l7
```

Nakon što smo nabavili kernel source, raspakiramo ga unutar /usr/src direktorija i pozicioniramo se u dobiveni direktorij /usr/src/linux-2.6.X.Y:

```
tar xzvf linux-2.6.X.Y.tar.gz

cd linux-2.6.X.Y
```

Zatim je potrebno patchirati kernel source:

```
patch -p1 < /usr/src/netfilter-layer7/kernel-2.6.X-layer7-Y.patch
```

Nakon toga, slijedi podešavanje kernela po želji (npr. pomoću make menuconfig metode - lokacije vrijede za verziju 2.6.18), uz obavezno uključenje sljedećih opcija:
- "Prompt for development and/or incomplete code/drivers" (pod "Code maturity level options")
- "Network packet filtering" (Networking › Networking options › Network packet filtering )
- "Netfilter Xtables support" (Networking › Networking options ›Network packet filtering › Core Netfilter Configuration)
- "Connection tracking" (Networking › Networking options › Network packet filtering › IP: Netfilter Configuration)
- "Connection tracking flow accounting" (u istom izborniku kao i prethodna opcija)
- "IP tables support" (u istom izborniku)
- "Layer 7 match support" (u istom izborniku)

Ostale Netfilter opcije nisu nužne, ali su poželjne (naročito FTP support).

Slijedi uobičajeno kompajliranje i instalacija kernela te restart računala:

```
make all

make modules_install

make install

mkinitrd -o /boot/initrd.img-2.6.18 2.6.18
```

uz podešavanje boot loadera (lilo/grub ili nešto treće).

Na redu je iptables. Prvo otpakiramo paket (npr. u /usr/src/iptables):

```
tar xzvf iptables-X.Y /usr/src/iptables

cd iptables
```

i zatim patchiramo iptables source:

```
patch -p1 < /usr/src/ netfilter-layer7/iptables-layer7-2.x.patch
```

Još je potrebno dodati pravo izvršavanja na datoteku ".layer7-test" unutar /root/iptables/extension direktorija:

```
chmod +x extensions/.layer7-test
```

Slijedi kompajliranje iptables-a:

```
make KERNEL_DIR=/putanja/do/patchiranog/kernela (u našem slu?aju /usr/src/linux-2.6.X
.Y)
```

i zatim instalacija (kao root):

```
make install KERNEL_DIR=/putanja/do/patchiranog/kernela (u našem slu?aju /usr/src/lin
ux-2.6.X.Y)
```

Za uspješnu instalaciju potrebno je imati već patchirani i konfigurirani kernel source.

Slijedi postavljanje definicija protokola koje je najbolje staviti u /etc/l7-protocols direktorij:

```
tar xzvf l7-protocols-YYYY-MM-DD.tar.gz /etc/l7-protocols
```

Moguća je njihova instalacija u proizvoljni direktorij, ali je za korištenje istih potrebno navesti njihovu lokaciju sa opcijom --l7dir.

Iz razloga što na paketu iproute2 nisu potrebne nikakve intervencije, dovoljno ih je instalirati:

```
apt-get update

apt-get install iproute
```

**Upotreba**
Sad ste spremni za rad. Moguće radnje su: blokiranje određenih protokola, kontroliranje pojasne širine i praćenje stanja na mreži. Svaka navedena radnja bit će opisana dalje u tekstu.

l7-filtar koristi standardnu iptables sintaksu, a osnovna sintaksa glasi:

```
iptables [tablica i lanac] -m layer7 --l7proto [ime_protokola] -j [akcija]
```

Iptables sintaksu možete naći na netfilter.org [3].

l7-filtar treba "vidjeti" sav mrežni promet koji želimo kontrolirati, što znači da promet treba proći pravila l7-filtra. To se postiže upotrebom POSTROUTING lanca u mangle tablici:

```
iptables -t mangle -A POSTROUTING -m layer7 --l7proto [itd.]
```

Napomena: ukoliko se koristi l7-filter verzija starija od 2.7, potrebno je ručno učitati ip_conntrack modul za kernel da bi l7-filtar ispravno radio. Novije verzije ga učitaju automatski.

**Blokiranje**
Blokiranje nije najpoželjniji način kontrole mrežnog prometa, i to iz više razloga:
- l7-filtar "matching" nije neotporan, tj. može se dogoditi da jedan protokol izgleda kao drugi (*false positive*)
- skoro svaka vrsta mrežnog prometa je legitimna (primjer su P2P protokoli koji se koriste za legalno razmjenjivanje i distribuciju besplatnih i slobodnih programa i dokumenata, a istovremeno se koriste za masovno kršenje autorskih prava)
Treba imati na umu da l7-filtar nije dizajniran s namjerom da se mrežni promet blokira, pa bi ovu radnju trebalo koristiti samo u nuždi.

**Kontrola pojasne širine**
Za kontrolu pojasne širine koristi se Netfilter za označavanje paketa (*mark*) i zatim se pomoću QoS (*Quality of Service*) tehnika može oblikovati promet označenih paketa. Samo označavanje radi se s opcijom

```
-j MARK --set-mark [integer]
```

dok se za oblikovanje koristi tc komandna linija koja je dio IPROUTE paketa. Slijedi primjer označavanja i filtriranja paketa koji koristi imap protokol:

```
iptables -t mangle -A POSTROUTING -m layer7 --l7proto imap -j MARK --set-mark 3
```

Vrijednost [integer] varijable je proizvoljna.

Oblikovanje mrežnog prometa tog označenog paketa se može izvoditi na sljedeći način:

```
tc filter add dev eth0 protocol ip parent 1:0 prio 1 handle 3 fw flowid 1:3
```

Time smo označeni imap promet usmjerili na treću podklasu prio metode za upravljanje redom čekanja (više o metodama za upravljanje slijedi kasnije u članku).
Komplicirana i nerazumljiva sintaksa tc komandne linije opisana je u LARTC HOWTO dokumentu, a za konkretnu upotrebu i lakši početak, u prilogu članka su dvije gotove skripte za oblikovanje mrežnog prometa. Jedna je skripta za premosnike (bridges), a jedna za "ne-premosnike" (non-bridges). Skriptu je potrebno modificirati na način da se odredi protokol koji se želi pratiti tj. oblikovati. Također, u slučaju ne-premosnika, potrebno je konfigurirati i NAT servis, a primjer NAT skripte je u prilogu. Skriptu treba modificirati na način da se promjeni IP adresa na kojoj se nalazi računalo koje obavlja NAT. Preporučljivo je da se skripte stave unutar /etc/init.d/ direktorija te se stave u startup proceduru sustava naredbom:

```
update-rc.d -f <ime_skripte> defaults
```

Podržan je priličan broj protokola, a to su uglavnom (nepoželjni) P2P protokoli te protokoli koje koriste računalne igre. Također je moguće napisati definicije za nove protokole, ukoliko za to postoji potreba. Uz protokole, moguće je definirati i tipove datoteka čiji promet želimo kontrolirati. Tako su podržane datoteke exe, gif, jpeg, pdf, rar, zip itd. Listu podržanih protokola i tipova datoteka možete naći pri kraju članka.

**Praćenje prometa na mreži**
Ako vas samo zanima kojim se intenzitetom koriste protokoli koje ste definirali unutar prve skripte, moguće je koristiti gornju naredbu, ali bez -j opcije. Na primjer:

```
iptables -t mangle -A POSTROUTING -m layer7 --l7proto imap
```

Statistika se u tom slučaju može pratiti pomoću naredbe

```
iptables  -t mangle -L -v
```

Već smo rekli da su metode upravljanja redovima algoritmi kojima se kontrolira način ulaska paketa u red i njihov izlazak. Ti algoritmi uključuju odlučivanje o tome koji se paketi propuštaju, kojim redoslijedom i brzinom, a to se radi unošenjem kašnjenja, preraspodjelom redoslijeda paketa i njihovim prioritiziranjem. Naravno, postoji više vrsta metoda za upravljanje, a u sljedećih nekoliko članaka sistematizirat ću osnovne metode upravljanja redovima čekanja i time olakšati odabir odgovarajuće metode ili više njih.

Prilozi:
Skripta za premosnike (bridges) [7]
Skripta za ne-premosnike (non-bridges) [8]
Skripta za NAT [9]

Linkovi:
l7-filter home page [10]
podržani protokoli i tipovi datoteka [11]
Linux Advanced Routing and Traffic Control [1]

- Logirajte [12] se za dodavanje komentara

čet, 2007-05-17 14:59 - Mirko Lovričević**Kuharice:** Za sistemce [13]
**Kategorije:** Mreža [14]
**Vote:** 0

No votes yet

**Source URL:** https://sysportal.carnet.hr/node/204

**Links**
[1] http://lartc.org
[2] https://kernel.org
[3] http://netfilter.org
[4] http://freshmeat.net/projects/iproute2
[5] http://sourceforge.net/project/showfiles.php?group_id=80085%20-%20netfilter-layer7-vX.Y.tar.gz
[6] http://sourceforge.net/project/showfiles.php?group_id=80085%20-%20l7-protocols-YYYY-MM-DD.tar.gz
[7] https://sysportal.carnet.hr/system/files/bridges.txt
[8] https://sysportal.carnet.hr/system/files/non-bridges.txt
[9] https://sysportal.carnet.hr/system/files/nat.txt
[10] http://l7-filter.sourceforge.net/
[11] http://l7-filter.sourceforge.net/protocols
[12] https://sysportal.carnet.hr/sysportallogin
[13] https://sysportal.carnet.hr/taxonomy/term/22
[14] https://sysportal.carnet.hr/taxonomy/term/29