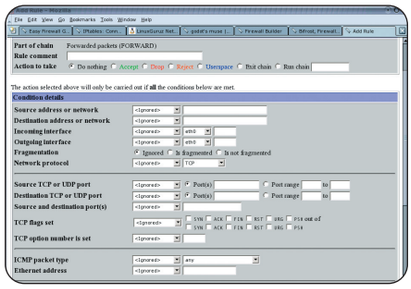


▲ Webmin, web sučelje za administraciju, ima jako detaljno riješen sustav konfiguracije iptablesa, no zahtijeva predznanje

nikakvu obavijest o onome što se dogodilo. Ta vrsta blokiranja zove se "pritajeno" (*stealth*) blokiranje i obično usporava hakera. Iako se ne računa kao odredište paketa, spomenimo i LOG opciju (-j LOG) koja će informaciju o paketu zapisati u log datoteku, ali mu se neće ništa dogoditi.

Dakle, sad bi potpuna naredba glasila "iptables -A INPUT -s 192.168.20.0/24 -j REJECT", a zabranjivala bi sve dolazne pakete iz dotične podmreže. Ako nismo zadovoljni s nekim pravilom, iz liste ga brišemo navodeći istu tu cjelovitu naredbu, ali s opcijom "-A" zamijenjenom s "-D". Popis svih pravila u filter tablici dobivamo s "iptables -L", a sve njih odjednom možemo obrisati s "iptables -F"

Definiranje uvjeta

Najvažniji dio cijele priče o filtriranju dobro je "hvatanje" paketa po raznim kriterijima. Dva najosnovnija uvjeta su izvorišne i odredišne (-d) adrese. To mogu biti pojedine IP adrese, mreže, pa i cijeli Internet (što je podrazumijevana vrijednost ako se ne navede). Možemo promatrati korišteni protokol (-p), poput TCP-a, UDP-a ili ICMP-a, pa i pojedini port koji oni koriste. Primjerice "-p tcp --dport 25" označava TCP promet i dolazni port 25. ICMP je informacijski servis Interneta i kao takav ne koristi portove već vrste prometa ("--icmp-type echo-request" označava ping pakete). Port ne mora biti napisan numerički, već po imenu servisa (kao u /etc/services).

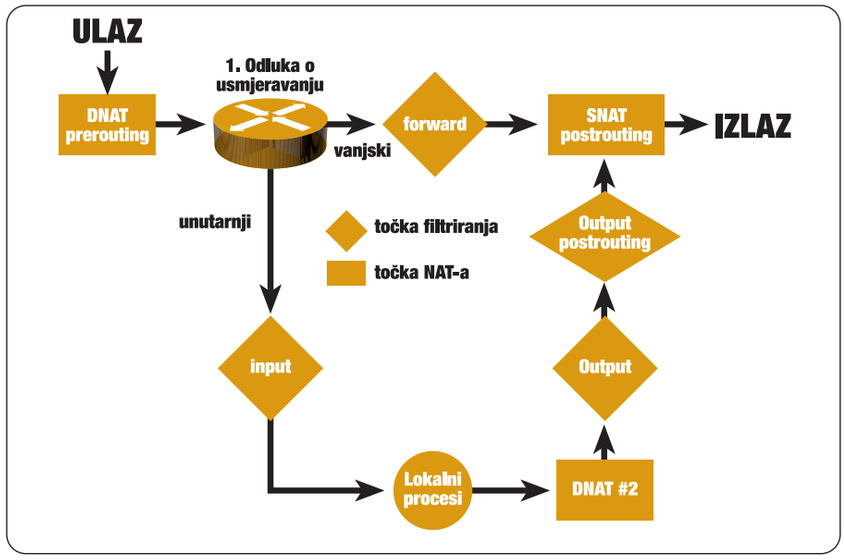
Pojedina pravila možemo i kombinirati, navodeći IP adrese, protokol i port, sve u jednom retku. Tipični primjer mogao bi biti "-p tcp --sport 22 -s 10.0.0.0/8 -d 1.2.0.0/16". Opcija "--sport" (*source port*) predstavlja port s kojeg paket dolazi. Rjeđe se koristi jer će većina mrežnih programa iskoristiti slučajno odabrani "visoki" (>1024) port za odlaznu komunikaciju.

Ne možemo a da (opet) ne spomenemo nmap, program za skeniranje udaljenih računala, koji se koristi tehnikama neispravnih TCP paketa kako bi lakše prolazio kroz *firewallove*. Netfilter ima odgovor i na takve trikove, te može zaustaviti bilo kakav paket pripremljen na taj način praćenjem raznih "zastavica" u njemu (--tcp-flags).

Značenje svakog od navedenih pravila može dobiti obrnuto značenje korištenjem uskličnika ("-s ! 192.168.20.1" označava sve pakete koji ne dolaze s te adrese).

Dodatna pravila

Osim osnovnih nabrojanih, zahvaljujući modularnoj strukturi netfiltera, moguće je koristiti i



▲ Standardni putovi prolaska paketa nazivaju se "lancima", a nama su bitni INPUT i OUTPUT

dodatne kriterije. Nazivaju se "match", a zajedničko im je da moraju biti eksplicitno pozvani po imenu modula (nalaze se u /lib/iptables). Jednostavan primjer filtriranja je po MAC adresi mrežne kartice (-m mac -mac-source 11:11:E1:A8:FF:11). To ima smisla samo u lokalnoj mreži, jer se sva računala izvan nje predstavljaju MAC adresom *gatewaya*.

Zanimljiva je mogućnost ograničavanje prometa po broju paketa koji pristižu u jedinici vremena. Pravilo "-m limit --limit 2/s -p tcp --dport 80" ograničit će broj zahtjeva prema web serveru na dva u sekundi.

Praćenje veza

Svakako najpopularnija mogućnost - po kojoj se netfilter najviše razlikuje od prethodnih verzija *firewalla* - je selekcija bazirana na trenutnom stanju veze. Ako smo uputili zahtjev prema nekom udaljenom računalu, svi paketi koje razmijenimo u bilo kojem smjeru bit će prepoznati kao dio te veze i tako propuštani, sve dok ona traje. Dinamički će se otvoriti samo uski prolaz za one pakete koji su nam bitni za komunikaciju započetu s naše strane. Razlikujemo *Invalid*, *New*, *Established* i *Related* vrstu veza, a za prethodno spomenutu funkcionalnost nama su potrebne posljednje dvije. Pretočeno u naredbu, to bi izgledalo ovako: "-m state --state ESTABLISHED, RELATED". "Related" se odnosi na složenije protokole, poput FTP-a, koji koristi više veza (kontrola + podaci) za komunikaciju.

Sučelja za konfiguraciju

Nakon što su protumačene osnove komunikacije i logika kojom se povodi netfilter, jednostavnije je korištenje raznih iptables sučelja. Postoje brojne skripte koje mogu olakšati konfiguraciju, no one često imaju svoju sintaksu kako bi prividno bilo olakšano korištenje, pa nam time samo daju dodatni materijal za učenje. Mnoge distribucije već prilikom instalacije podignu neki stupanj zaštite. Za potpuniju kontrolu, brojnije opcije i ugodniji rad mogu se koristiti neki od malobrojnih grafičkih sučelja. Veći popis (i brojne resurse za netfilter) može se naći na **www.linuxguruz.com/iptables**. Postoje i razna web sučelja, a nama se zbog jednostavnosti svidio **easyfwgen.morizot.net/gen**.

Konfiguracija kojom smo zadovoljni sprema se naredbom "iptables-save > datoteka", dok se iz nje može izvući s "iptables-restore < datoteka". Kod *restarta* računala sva ručno dodana pravila će se obrisati, pa je preporučljivo na ovaj način spremiti promjene, te ih naknadno učitati.

Primjeri

Najosnovniji - a ujedno vrlo siguran - *firewall* može imati tek par redaka:

```
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -A INPUT  -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A OUTPUT -m state --state NEW,ESTABLISHED -j ACCEPT
iptables -A INPUT -p tcp --dport 22 -m state --state NEW,ESTABLISHED -j ACCEPT
```

Par primjera cjelovitih naredbi, u tekstu skraćenih:

```
iptables -A INPUT -s 192.168.20.30 -j REJECT --reject-with tcp-reset
iptables -A INPUT -p tcp --dport 25 -j ACCEPT
iptables -A FORWARD --tcp-flags ALL SYN,RST,ACK,FIN,URG -j DROP
iptables -D INPUT -p udp --sport 137:139 -j DROP
iptables -A INPUT -p tcp --sport 22 -s ! 10.0.0.0/8 -d 1.2.0.0/16 -j REJECT
iptables -A OUTPUT -p tcp --dport 80 -m limit --limit 2/s -j LOG
iptables -A OUTPUT -p icmp --icmp-type echo-request -j DROP
```