OpenBSD

PF: Primer: Zastitni Zid za Kucnu mrezu ili Malu Kancelariju

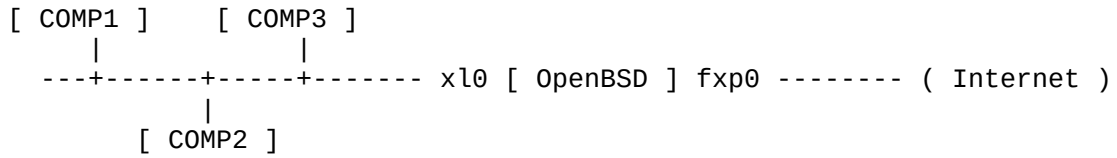
Sarzaj

Scenario

U ovom primeru, PF radi na OpenBSD masini predstavljajuci zastitni zid i NAT gateway za malu mrezu u kuci ili kancelariji. Glavni cilj je da se omoguci Internet pristup mrezi i da se dozvoli ogranceni pristup do masine sa zastitnim zidom sa Interneta, i omoguci pristup unutrasnjem web serveru sa Interneta. Ovaj document ce proci kroz kompletan skup pravila koja rade upravo to.

Mreza

Mreza je ovako podesena:



Postoji nekoliko kompjutera na unutrasnjoj mrezi; diagram prikazuje tri ali je u stvari pravi broj nevazan. Ovi kompjuteri su regularne radne stanice koje se koriste za surfovanje web-om, email, chat-ovanje, itd., osim COMP3 na kom je podignut mali web server. Unutrasnja mreza koristi 192.168.0.0 / 255.255.255.0 mrezni blok.
OpenBSD zastitni zid je Celeron 300 sa dve mrezne kartice: 3com 3c905B (xl0) i Intel EtherExpress Pro/100 (fxp0). Zastitni zid je kablom povezan do Interneta i koristi NAT da deli ovu konekciju sa unutrasnjom mrezom. IP adresa na spoljnjem interfejsu je dinamicki dodeljena od strane Internet Servis Provajdera.

Cilj

Ciljevi su:

- Omoguci neograniceni prostup Internetu svakom unutrasnjem kompjuteru.
- Koristi "default deny" filtriranje skupa pravila.
- Dozvoli sledeci dolazni saobracaj od Interneta do zastitnog zida:
  - SSH (TCP port 22): ovo ce se koristiti za spoljno odrzavanje masine sa zastitnim zidom.
  - Auth/Ident (TCP port 113): koriscen od nekih servisa kao sto su SMTP i IRC.
  - ICMP Echo Requests: ICMP tip paketa koje koristi ping(8).
- Preusmeri TCP port 80 pokusaje konektovanja (pokusaji da se pristupi web serveru) do kompjutera COMP3. Isto tako, dozvoli TCP port 80 saobracaj namenjen COMP3 kroz zastitni zid.
- Logiraj stastistike filtriranja na spoljasnjem interfejsu.
- Podrazumevano, odgovori sa TCP RST ili ICMP Unreachable za blokirane pakete.
- Napravi skup pravila jednostavnijim i lakim za odrzavanje koliko je to moguce.

Priprema

Ovaj document pretpostavlja da je OpenBSD host odgovarajuce konfigurisan da radi kao router, ukljucujuci proveravanje IP mreznog podesavanja, Internet konekcije, i podesavanja net.inet.ip.forwarding na "1". Verovatno cete i aktivirati PF u /etc/rc.conf.local.

Skup Pravila

U nastavku cemo korak-po-korak procu kroz skup pravila koji ce postici gore navedene ciljeve.

Makroi

Sledeci makroi su definisani da olaksaju odrzavanje i citanje skupa pravila.

```
ext_if="fxp0"
int_if="xl0"

tcp_services="{ 22, 113 }"
icmp_types="echoreq"

comp3="192.168.0.3"
```

Prve dve linije definisu mrezne interfejse na kojima ce se desavati filtriranje. Definisuci ih ovde, ako trebamo premestiti ovaj sistem na drugu masinu sa razlicitim hardverom, mozemo promeniti samo te dve linije, i ostatak skupa pravila se i dalje moze koristiti. Treca i cetvrta linija izlistavaju TCP brojeve portova servisa koji ce biti otvoreni ka Internetu (SSH i ident/auth) i ICMP tip paketa koji ce biti prihvaceni od zastitnog zida. Na kraju, zadnja linija definise IP adresu COMP3.

Beleska: Ako Internet veza zahteva PPPoE, onda filtriranje i NAT mora da se radi na tun0 interfejsu a *ne* na xl0.

Opcije

Sledece dve opcije ce postaviti podrazumevani odgovor za block filter pravila i “ukljuciti’ logiranje statistike za spoljni interfejs:

```
set block-policy return
set loginterface $ext_if
```

Svaki Unix sistem ima "loopback" interfejs. To je virtualni mrezni interfejs koji koriste aplikacije za medjusobno komuniciranje unutar sistema. Na OpenBSD-u, loopback interfejs je lo(4). Smatra se da je najbolje da se onemoguce sva filtriranja na loopback interfejsu. Koristeci set skip mozemo postici ovo.

```
set skip on lo
```

Primetite da 'preskacemo' celu interfejs grupu lo, na ovaj nacin, u slucaju da kasnije dodamo dodatne loopback interfejse, necemo morati da menjamo ovaj deo naseg postojeceg fajla sa pravilima.

Scrub

Ne postoji razlog za ne koriscenje preporucenog 'ciscenja' svog dolaznog saobracaja, stoga je dovoljno staviti samo jednu liniju:

```
scrub in
```

Mrezno Prevodjenje Adresa

Da bi ste koristili NAT za celu unutrasnju mrezu koristi se sledece nat pravilo:

```
nat on $ext_if from !($ext_if) to any -> ($ext_if)
```

Ovo "!($ext_if)" se lako moze zameniti sa "$int_if" u ovom slucaju, ali ako ste dodali nekoliko mreznih interfejsa, morate dodati dodatna NAT pravila, dok u ovom slucaju, NAT-om ce se rukovoditi na svim zasticenim interfejsima.
Posto je IP adresa na spoljnom interfejsu dinamicki dodeljena, zagradama se okruzuje interfejs na kom se desava prevodjenje tako da ce PF primetiti kada se adresa promeni.
Posto zelimo da nam FTP proxy radi, stavicemo i NAT anchor:

```
nat-anchor "ftp-proxy/*"
```

Preusmeravanje

Prva pravila preusmeravanja koja su nam potrebna su za ftp-proxy(8) tako da FTP klijenti na lokalnoj mrezi mogu da se povezu na FTP servere na Internetu.

```
rdr-anchor "ftp-proxy/*"
rdr on $int_if proto tcp from any to any port 21 -> 127.0.0.1 port 8021
```

Primetite da ce ovo pravilo hvatati FTP konekcije na port 21. Ako se korisnici regularno konektuju na FTP servere na drugim portovima, onda treba da se koristi lista za odredjivanje odredisnih portova, na primer: from any to any port { 21, 2121 }.
Zadnje pravilo za preusmeravanje hvata sve pokusaje nekog na Internetu da se poveze na TCP port 80 na zastitnom zidu. Legitimni pokusaji da se pristupi ovom portu ce biti od strane korisnika koji pokusavaju da pristupe web serveru na mrezi. Ovi pokusaju konektovanja se trebaju preusmeriti na COMP3:

```
rdr on $ext_if proto tcp from any to any port 80 -> $comp3
```

Filter Pravila

A sada pravila za filtriranje. Pocenite sa podrazumevanim odbijanjem pristupa:

```
block in
```

Sada ce sav saobracaj koji pokusava da dodje na interfejs biti blokiran, cak i onaj sa unutrasnje mreze. Pravila koja slede ce otvoriti zastitni zid prema gore navedenim ciljevima kao i sve neophodne virtualne interfejse. Imajte na umu, pf moze da blokira saobracaj koji dolazi ili odlazi sa interfejsa. Mozete olaksati sebi ako odaberete da filtrirate saobracaj u jednom pravcu, radije nego da pokusavate da sredjujete stvari kada filtrirate neke dolazne stvari, i neke odlazne stvari. U nasem slucaju, usredsredicemo se na filtriranje dolaznog saobracaja, ali kada saobracaj vec dodje do interfejsa, necemo pokusavati da sprecimo njegov odlazak, tako da cemo uraditi sledece:

```
pass out keep state
```

Potreban nam je anchor za ftp-proxy(8):

```
anchor "ftp-proxy/*"
```

Dobro je da se koristi zastita od laznih adresa:

```
antispoof quick for { lo $int_if }
```

Sada otvorite portove koje koriste oni mrezni servisi koji ce biti dostupni Internetu. Prvo, saobracaj koji je namenjen samom firewall-u:

```
pass in on $ext_if inet proto tcp from any to ($ext_if) \
   port $tcp_services flags S/SA keep state
```

Odredjivanje mreznih portova u makrou $tcp_services olaksava da se otvore dodatni servisi ka Internetu tako sto cete jednostavno editovati makro i ponovo ucitati skup pravila. i UDP servisi se mogu otvoriti kreiranjem $udp_services makroa i dodavanjem filter pravila, slicnim gore navedenim, koji odredjuje proto udp.
Posto imamo rdr pravilo koje dozvoljava web server saobracaj to COMP3, MORAMO da dozvolimo i ovaj saobracaj kroz zastitni zid:

```
pass in on $ext_if inet proto tcp from any to $comp3 port 80 \
    flags S/SA synproxy state
```

Za malo dodatne sigurnosti, iskoristi cemo TCP SYN Proxy da bi jos vise zastitili web server.
ICMP saobracaj treba biti propusten:

```
pass in inet proto icmp all icmp-type $icmp_types keep state
```

Slicno $tcp_services makrou, i $icmp_types makro se lako moze editovati da bi se promenio tip ICMP paketa kojima ce se dozvoliti da dodju do zastitnog zida. Primetite da se ovo pravilo primenjuje na sve mrezne interfejse.
Sada se mora propustiti saobracaj do, i od unutrasnje mreze. Pretpostavicemo da korisnici na unutrsanjoj mrezi znaju sta rade i nece praviti probleme. Ovo ne znaci da je pretpostavka tacna; stroziji skup pravila ce verovatno vise odgovarati za mnoga oruzenja.

```
pass in quick on $int_if
```

TCP, UDP, i ICMP saobracaju je dozvoljeno da napusti zastitni zid prema Internetu zbog ranije "pass out keep state" linije. Informacije o konekciji se cuvaju tako da se povratni paketi prosledjuju kroz zastinti zid.

Kompletan Skup Pravila

```
# macros
ext_if="fxp0"
int_if="xl0"

tcp_services="{ 22, 113 }"
icmp_types="echoreq"

comp3="192.168.0.3"

# options
set block-policy return
set loginterface $ext_if

set skip on lo

# scrub
scrub in

# nat/rdr
nat on $ext_if from !($ext_if) -> ($ext_if:0)
nat-anchor "ftp-proxy/*"
rdr-anchor "ftp-proxy/*"

rdr pass on $int_if proto tcp to port ftp -> 127.0.0.1 port 8021
rdr on $ext_if proto tcp from any to any port 80 -> $comp3

# filter rules
block in

pass out keep state

anchor "ftp-proxy/*"
antispoof quick for { lo $int_if }

pass in on $ext_if inet proto tcp from any to ($ext_if) \
   port $tcp_services flags S/SA keep state

pass in on $ext_if inet proto tcp from any to $comp3 port 80 \
    flags S/SA synproxy state

pass in inet proto icmp all icmp-type $icmp_types keep state

pass quick on $int_if
```

~Dalibor Gudzic@soxxx