

# Tűzfalak építése ipsettel

Kadlecsik József

<[kadlec@blackhole.kfki.hu](mailto:kadlec@blackhole.kfki.hu)>

KFKI RMKI

# Tartalom

- Miért és hol van rá szükség
- ipset
- ipset és iptables

# Speciális tűzfalak

- Nagyon magas szabálysám
  - Gyors kiértékelő algoritmus
- Gyakran változó szabályok
  - Gyorsan változtatható szabálytárolási struktúra
- Kis fizikai memóriával rendelkező gépek
  - Kis helyigényre optimalizált tárolás

# Speciális tűzfalakra: iptables?

- iptables: <http://www.netfilter.org/>
- Magas szabálysám esetében lassú
  - Lineáris szabályfeldolgozás
- Gyakori szabályváltoztatáskor lassú
  - A kernel-userspace között az **összes** szabály oda-vissza utazik **egyetlen** új szabály hozzáadásakor is: szabálytárolási forma a blob
- Nincs kiemelkedő memória igénye

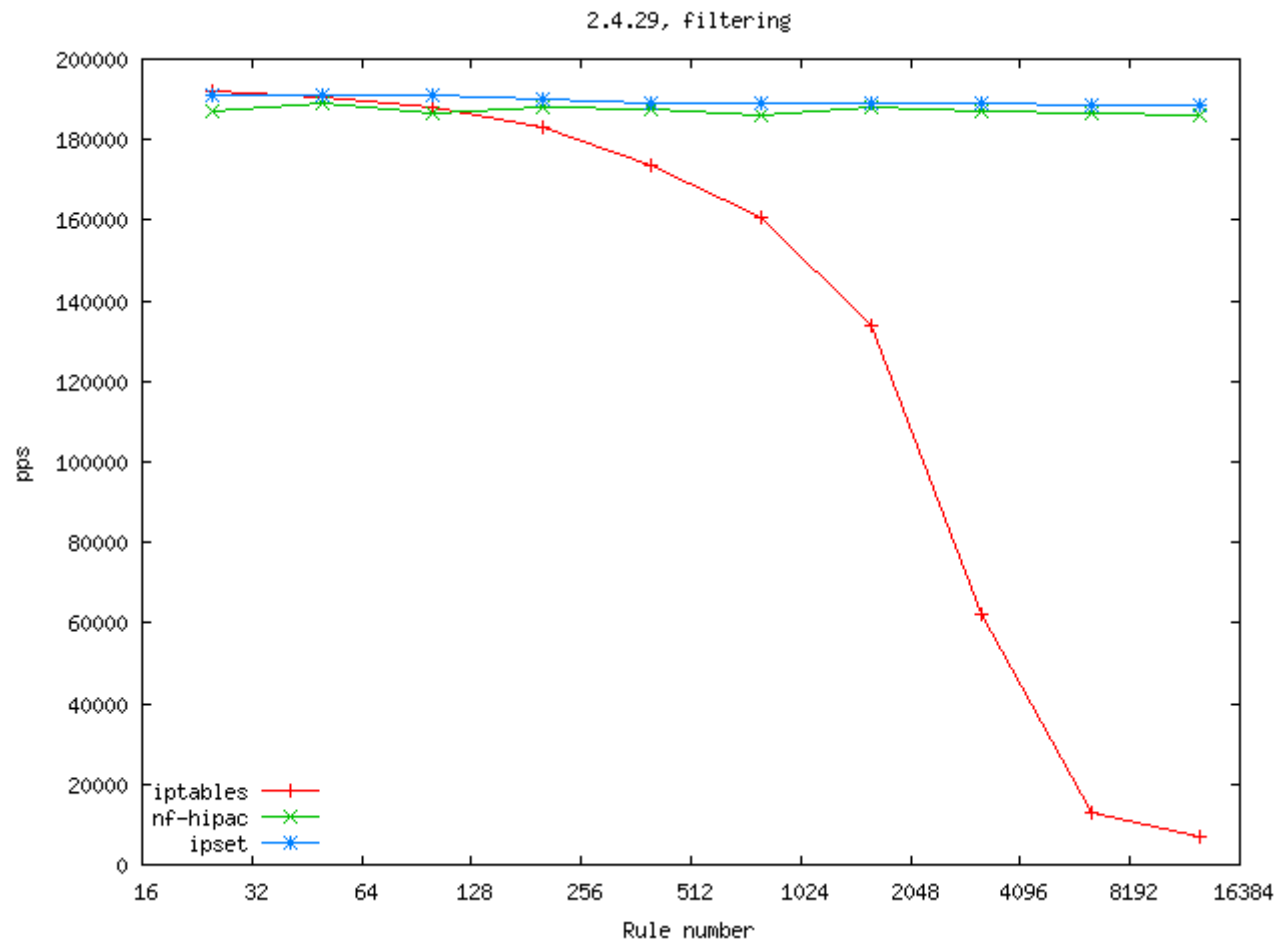
# Speciális tűzfalakra: nf-hipac?

- nf-hipac: <http://www.hipac.org/>
- Magas szabálysám esetében is gyors
  - Komplex kiértékelő algoritmus
- Gyakori szabályváltoztatáskor is gyors
  - A kernel-userspace között csak az új szabály utazik, speciális tárolási forma
- Memória igénye nem ismert/nem kiemelkedő

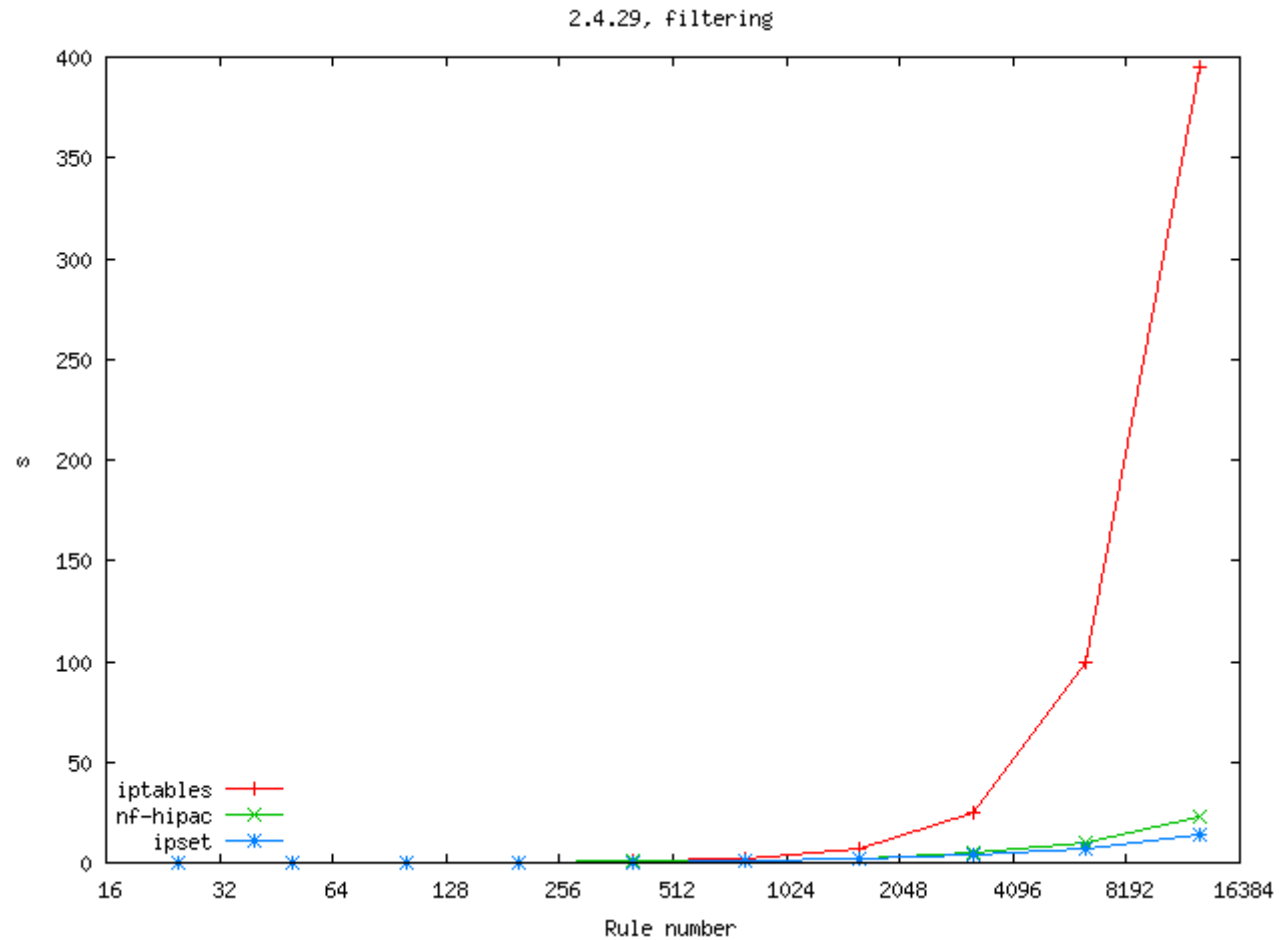
# Speciális tűzfalakra: ipset?

- ipset: <http://ipset.netfilter.org/>
- Magas szabálysám esetében is gyors
  - Egyszerű kiértékelő algoritmusok
- Gyakori szabályváltoztatáskor is gyors
  - A kernel-userspace között csak az új szabály utazik, egyszerű tárolási forma
- Memória igénye kicsi

# iptables, nf-hipac és ipset I.



# iptables, nf-hipac és ipset II.



# Az ipset és ippool

- Joakim Axelsson: ippool, bitmap típus
- Joakim Axelsson, Patrick Schaaf és Martin Josefsson: moduláris ippool, bitmap és macipmap típusok
- ipset: részben újraírt ippool több új típussal

# ipset

- Adathalmazok a kernelben, amelyek IP címet, portszámot, MAC címet vagy ezek kombinációit tárolhatják
- Különböző halmaz-típusok: map, hash, tree
- Speciális program a halmazok kezelésére: `ipset`
- A halmazokra iptables szabállyal hivatkozhatunk: `match` és `target`
- [Halmazok elemei más halmazokhoz köthetők: `binding`]

# ipmap (bitmap) típus

- Tartományba eső IP címek tárolására
  - egy IP címet egy **biten** tárolunk
- Maximálisan 65536 elem
- Tipikusan általános hozzáférés-engedélyezés (kik férhetnek hozzá az Internethez) és tiltás (kik vannak explicit kitiltva)

# ipmap példák

- IP címek tárolása:

```
ipset -N set1 ipmap --from 192.168.0.0 --to 192.168.255.255
```

```
[ipset -N set1 ipmap --network 192.168.0.0/16]
```

```
ipset -A set1 192.168.0.1
```

```
ipset -A set1 192.168.0.5
```

- Vagy azonos méretű (al)hálózatok:

```
ipset -N set2 ipmap --network 0.0.0.0/0 --netmask 16
```

```
ipset -A set2 10.1.0.0
```

```
ipset -A set2 10.7.0.1
```

# macipmap típus

- IP és MAC cím párosának tárolására
  - 8 byte
- Csak forrás MAC cím
- Első egyezés-kereséskor MAC cím automatikusan kitölthető

```
ipset -N macipset1 macipmap --from 192.168.0.0 \  
--to 192.168.255.255
```

```
ipset -A macipset1 192.168.1.1%00:01:23:45:67:89
```

```
ipset -A macipset1 192.168.2.3
```

# portmap típus

- Portszoámok tárolására
  - Minden portot egy biten tárolunk

```
ipset -N portset1 portmap --from 0 --to 1024
```

```
ipset -A portset1 22
```

# iphash típus

- Random IP címeket fix méretű (hashsize) hashben tároluk
  - Hashelés megismétlése (probes) ütközés esetén
  - Hash növelés, ha megtelt (resize)
- Optimalizálhatunk sebességre:
  - nagy hashsize, kis probes és nagy resize paraméterek
- Vagy memóriahasználatra:
  - Kis hashsize, nagy probes és kis resize paraméterek

# iphash példák

- IP címek tárolása

```
ipset -N myhash1 iphash --hashsize 1024 --probes 2 --resize 50
```

```
ipset -A myhash1 10.1.1.1
```

```
ipset -A myhash1 192.168.52.3
```

- Azonos méretű netblokkok:

```
ipset -N myhash2 iphash --hashsize 64 --probes 8 --resize 0 \  
--netmask 24
```

```
ipset -A myhash2 10.1.1.0
```

```
ipset -A myhash2 192.168.52.0
```

# nethash típus

- Hasonló az iphash-hez, de /1 és /31 közötti különböző méretű netblokkok tárolhatók benne
  - Hálózati cím és netmask összesen 32 biten

```
ipset -N hash3 nethash --hashsize 1024 --probes 2 --resize 50
```

```
ipset -A hash3 192.168.1.0/24
```

```
ipset -A hash3 10.1.8.0/21
```

# Hálózatok tárolása

- Azonos méretű alhálóok egy hálózatból: ipmap

```
ipset -N map1 ipmap --network 192.168.0.0/16 --netmask 24
```

```
ipset -A map1 192.168.1.0/24
```

```
ipset -A map1 192.168.2.0/24
```

- Azonos méretű alhálóok, nem egy hálózatból:  
iphash

```
ipset -N hash1 iphash --netmask 24
```

```
ipset -A hash1 10.1.1.0/24
```

```
ipset -A hash1 192.168.2.0/24
```

- Különböző méretű alhálóok: nethash

```
ipset -N hash2 nethash
```

```
ipset -A hash2 192.168.1.0/24
```

```
ipset -A hash2 10.1.8.0/21
```

# ipporthash típus

- Max /16-os hálózatba eső IP címek és port párosok tárolására
  - Egy IP cím + port párost 32 biten tárol
- Szolgáltatások lefedésére

```
ipset -N porthash ipporthash --network 192.168.0.0/16 \  
      --hashsize 1024 --probes 2 --resize 5
```

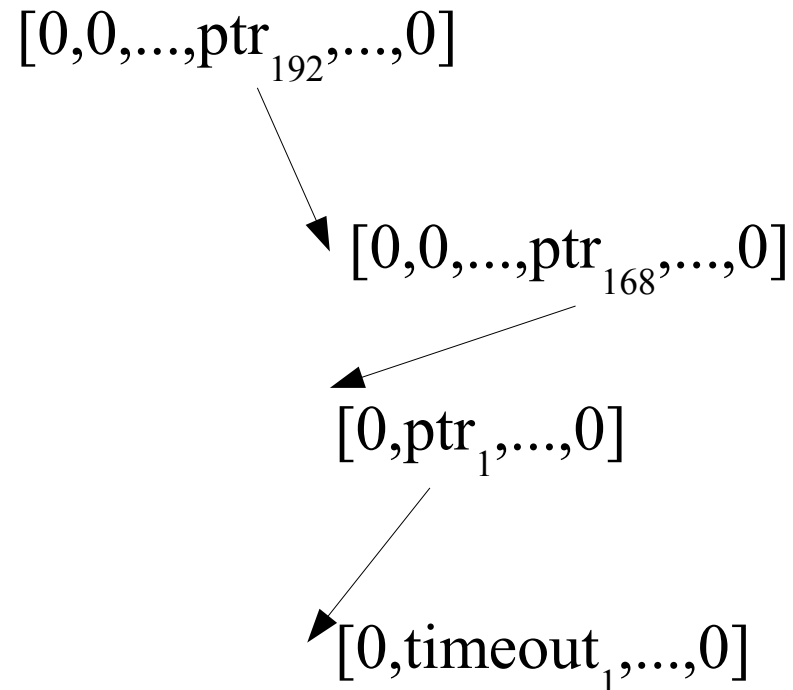
```
ipset -A porthash 192.168.1.1:22
```

```
ipset -A porthash 192.168.1.1:80
```

```
ipset -A porthash 192.168.1.4:25
```

# iptree típus

- IP címek fa struktúrában való tárolása, automatikus timeout támogatással: 192.168.1.1



# iptree példa

```
ipset -N tree1 iptree --timeout 600
```

```
ipset -A tree1 192.168.1.1:300
```

```
ipset -A tree1 192.168.2.8:0
```

# iptreemap típus

- Hasonló az iptree-hez, de az utolsó octetet bitmap-ben tárolja
- Elemek hozzáadásakor IP címet, tartományt vagy hálózatot is hozzá lehet adni:

```
ipset -N tree2 iptreemap --timeout 600
```

```
ipset -A tree2 192.168.1.1
```

```
ipset -A tree2 192.168.2.0/24
```

```
ipset -A tree2 192.168.3.1-192.168.3.18
```

# iptables set match

- Hogyan keresünk egyezést egy ipset halmazban iptables-ből: set match
- `-m set <setname> src|dst[,src|dst]`

```
ipset -N servers ipporthash --network 192.168.0.0/24
```

```
ipset -A servers ipporthash 192.168.0.1:25
```

```
ipset -A servers ipporthash 192.168.0.2:80
```

```
iptables -A FORWARD -m set --set servers dst,dst \  
-m state --state NEW -j log-accept
```

# SET target

- Iptables segítségével is hozzáadhatunk elemeket egy ipset halmazhoz
- `-j SET --add-set|--del-set <setname> src|dst[,...]`

```
ipset -N spammers iptree --timeout $((60*60*24*7))
```

```
iptables -A FORWARD -d <honeypot> -p tcp --dport 25 \  
-j SET --add-set spammers src
```

```
iptables -A FORWARD -p tcp --dport 25 \  
-m set --set spammers src -j TARPIT
```

# ipset halmazokat elmenthetünk és visszatölthetünk

- Iptables-save-hez hasonló szintaxis, szigorú sorrenddel a mentési fájlban

```
ipset -S > ipset.rules
```

```
ipset -R < ipset.rules
```

# Swap halmazokra

- Iptables-el hivatkozott ipset halmaz nem törölhető – de bármikor kicserélhető egy másik (létező) halmazra:

```
ipset -N main-set ...
```

```
iptables -A ... -m set --set main-set ...
```

```
ipset -N main-set-new ...
```

```
ipset -W main-set main-set-new
```

# ipsettel fölépített tűzfal I.

```
# Összes belső kliens, amelyik használhatja az Internetet
# (lehetne macipmap típus is, ha mind a LAN-on van):
ipset -N kliensek ipmap --network 192.168.0.0/16
ipset -A kliensek 192.168.10.1
...
# Összes belső szerver, amelyek az Internet felé
# szolgáltatnak
ipset -N szerverek ipporthash --network 192.168.0.0/16
ipset -A szerverek 192.168.0.1:22
ipset -A szerverek 192.168.0.2:25
...
```

# ipsettel fölépített tűzfal II.

```
# A tűzfal szabályok: stateful
iptables -A FORWARD -m state --state ESTABLISHED,RELATED -J ACCEPT

# Naplózási láncok
...

# Szerverekre vonatkozó szabályok
iptables -A FORWARD -m set --set szerverek dst,dst \
          -m state --state NEW -J log-accept
# Kliensekre vonatkozó szabályok
iptables -A FORWARD -m set --set kliensek src \
          -m state --state NEW -J log-accept
# Egyébként naplózunk mindent és tiltunk
iptables -A FORWARD -j log-drop
```

# Teendők

- Hiányzó típusok implementálása:
  - ipportip hármás
  - union
- IPv6 támogatás
- Sockopt helyett netlink interfész
- Kerüljön be a mainline kernelbe :-)

Köszönöm a figyelmet!