Osobní firewall s iptables

Ondřej Caletka

O.Caletka@sh.cvut.cz
http://www.pslib.cz/caletka

Osobní firewall s iptables

- Rychlokurz síťování z příkazového řádku
- Jak prochází paket počítačem
- Rychlokurz obsluhy programu iptables
- Jednoduché nastavení stavového firewallu
- Jak funguje protokol FTP
- Nastavení NATu

Rychlokurz síťování

- Nastavení z DHCP: *dhcpcd eth0*
- Ruční nastavení:

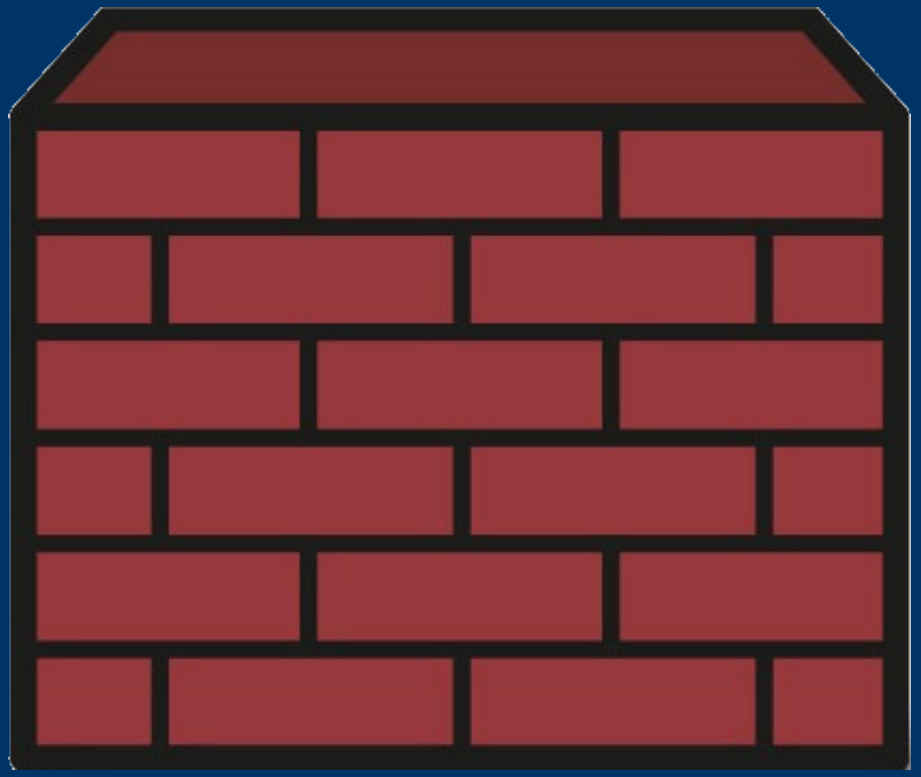
```
ip addr add dev eth0 192.168.0.25/24
ip route add default via 192.168.0.1
```

- Nebo postaru:

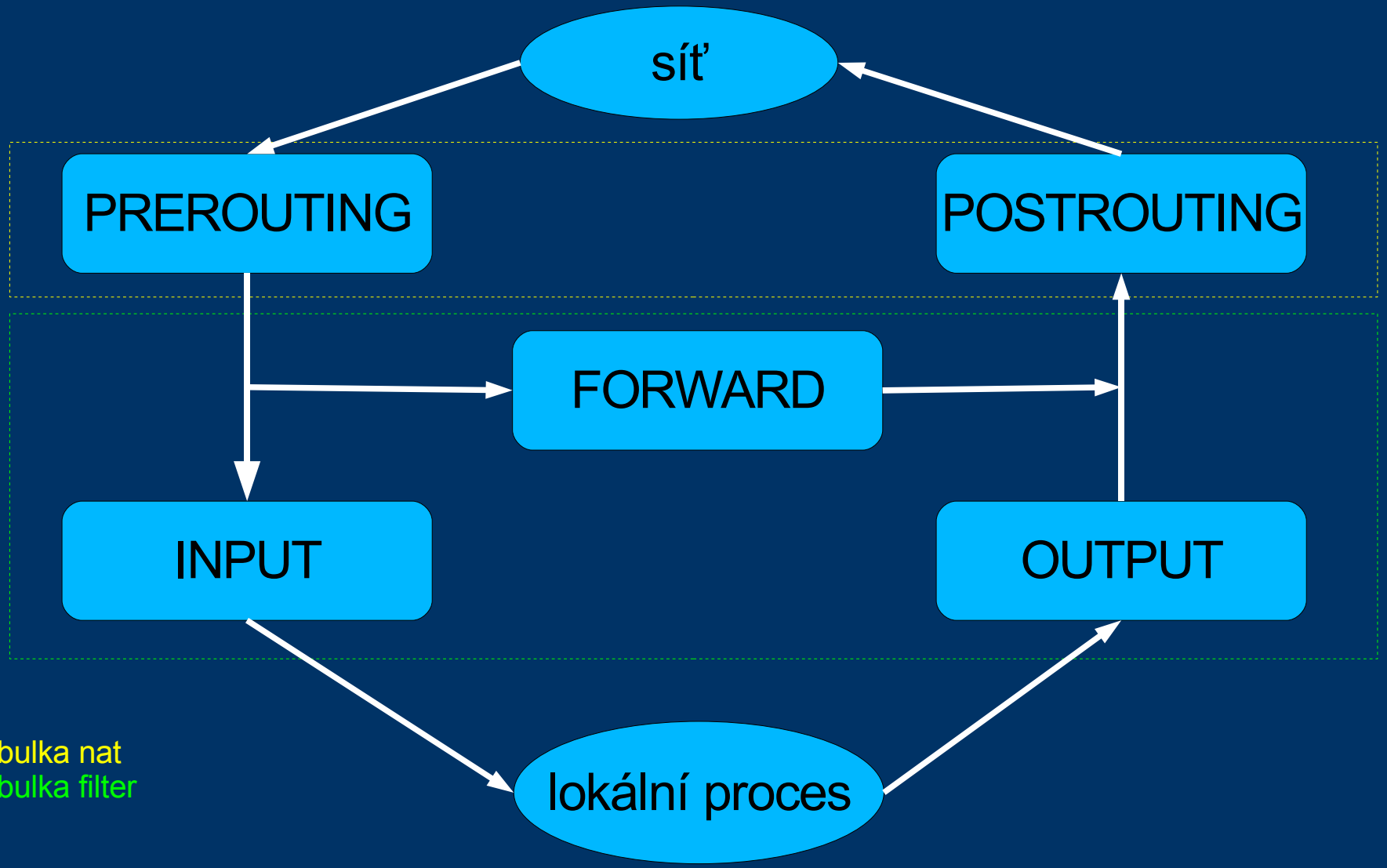
```
ifconfig eth0 up 192.168.0.25 netmask \
  255.255.255.0
route add default gw 192.168.0.1
```

Co je to firewall

- požární stěna :-)
- netfilter v 2.4 a 2.6
- ipchains v 2.2
- ipfwadm v 2.0
- ...



Jak prochází paket počítačem



Tabulka nat
Tabulka filter

Rychlokurz obsluhy iptables

```
iptables -t <tabulka> -A INPUT <podmínky> -j <cíl>
```

- <tabulka> nabývá hodnot filter, nat, mangle a není-li uvedeno, počítá se s tabulkou filter
- Následuje řídicí přepínač – ten je JEDINÝ velkým písmenem:

```
-A INPUT -> Append - přidat na konec řetězu INPUT
-I INPUT 1 -> Insert - vložit před první pozici
-R INPUT 1 -> Replace - zaměnit pravidlo číslo 1
-D INPUT 1 -> Delete - vymazat pravidlo číslo 1
-F INPUT -> Flush - vyprázdní řetěz INPUT
-L INPUT -> List - vypíše obsah řetězu INPUT
-P INPUT DROP -> Policy - nastaví politiku řetězu na DROP
```

- Jak vypsat podrobný seznam pravidel?

```
# iptables -t <tabulka> -L [řetěz] -v --line-numbers
```

Jak fungují pravidla?

Příklad

```
iptables -A INPUT -i eth0 -s \! 10.0.0.1 -j ACCEPT
```

vytvoří jeden řádek v řetězu INPUT.

Pokud bude splněno:

- Vstupní rozhraní je eth0

A ZÁROVEŇ

- Zdrojová IP adresa NENÍ 10.0.0.1

pak se provede skok (-j[ump]) na cíl ACCEPT.

Toto je zrovna speciální cíl, který způsobí zastavení vyhodnocování pravidel a povolení průchodu paketu.

Pravidla o pravidlech

- Jednotlivá pravidla se procházejí postupně od prvního k poslednímu a to obecně i v případě, je-li nalezena shoda
- Výjimku tvoří pravidla s cílem ACCEPT, DROP, nebo REJECT, které zastaví vyhodnocování dalších pravidel
- Pokud nevyhoví ani jedno pravidlo v řetězu, uplatní se výchozí politika řetězu, která je u prázdného netfiltru nastavena na ACCEPT

Přehled standardních podmínek

```
-s 10.0.0.0/24 -> zdrojová (source) adresa 10.0.0.*
-d 10.20.32.1 -> cílová (destination) adresa 10.20.32.1
-i eth+ -> přichází po rozhraní eth*
-o \! ppp2 -> neměl by odcházet rozhraním ppp2
-p tcp -> protokol tcp
--dport 80 -> cílový port 80 (platí pouze současně s -p)
```

- Rozšířené podmínky („moduly“) - aktivují se -m <podmínka>

```
-m state --state NEW -> nová spojení
-m state --state ESTABLISHED,RELATED
  -> stávající spojení a spojení k stávajícím vztažená
-m state --state INVALID -> neplatná spojení

-m limit --limit 3/second --limit-burst 5
  -> vyhoví pro 5 paketů v řadě a pro další jen nejvýše
  třikrát za sekundu
```

Přehled standardních cílů

- V tabulce filter:

```
-j ACCEPT -> paket je propuštěn, procházení ukončeno
-j DROP -> paket je v tichosti zahozen, procházení ukončeno
-j REJECT -> jako DROP, navíc odesílateli je poslána
  chybová zpráva ICMP – destination port unreachable

-j LOG -> zaznamená průchod paketu přes syslog
```

- V tabulce mangle:

```
-j MARK -> orazítkuje paket (pro shaping...)
```

- V tabulce nat:

```
-j SNAT --to-source 194.35.24.51 -> source NAT - změní
  zdrojovou adresu IP paketu na zadanou a zařídí i zpětné
  přemapování.

-j DNAT –to-destination 10.0.0.10 -> destination NAT –
  změní cílovou adresu na uvedenou.
```

Dodatek

Kromě standardních řetězů je možné vytvořit libovolně množství dalších řetězů a do nich posílat pakety příkazem -j už předfiltrované. To se hodí třeba, když chceme stejná pravidla uplatnit pro INPUT i FORWARD řetěz.

U osobního firewallu to ale není zapotřebí.

Firewall

Včasné zavolání může zachránit životy

TÍSŇOVÉ VOLÁNÍ / EMERGENCY CALL / NOTRUF

112

Jste-li svědkem nebo účastníkem mimořádné události,
volejte tísňovou linku 112

- zachovejte klid
- v rámci svých možností poskytněte pomoc ostatním
- dbejte pokynů záchranářů a policistů

Sdělení Ministerstva vnitra ČR:
Současně zůstávají v platnosti národní čísla tísňového volání – 150 hasiči, 155 zdravotnická záchranná služba, 158 policie a 156 městská policie.

Osobní firewall – požadavky

- Budeme filtrovat pouze příchozí spojení
- Pustíme dovnitř loopback (!!!)
- Pustíme dovnitř všechna spojení, která u nás začala, nebo která s nimi souvisí (např. FTP data).
- Pustíme dovnitř spojení na vyjmenované porty (SSH)
- Pustíme dovnitř ICMP pakety (!), ale jejich množství omezíme (Ping Of Death)
- Pustíme dovnitř „VLC“, tj. UDP multicast s cílovým portem 1234 (video) a 9875 (SAP)
- Požadavky na ident zdvořile odmítneme (REJECT)

Osobní firewall - realizace

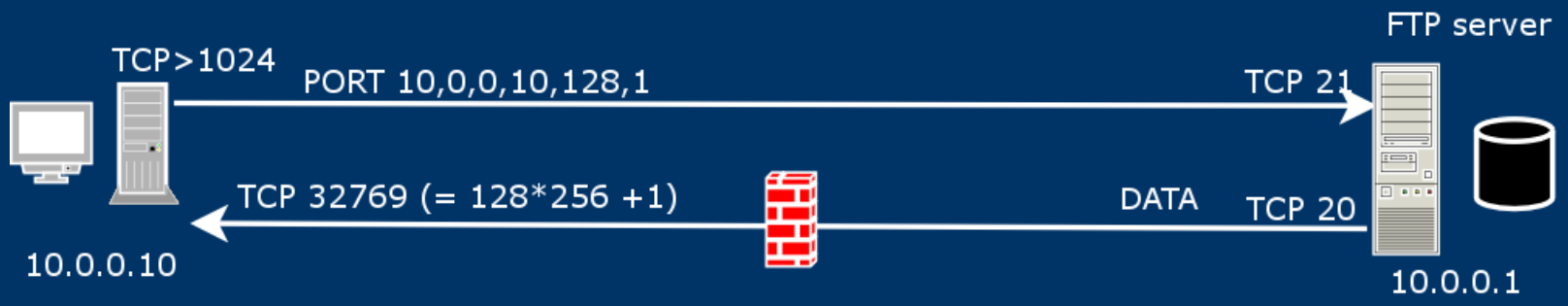
```
iptables -A INPUT -i lo -j ACCEPT
iptables -A INPUT -state ESTABLISHED,RELATED \
                  -j ACCEPT
iptables -A INPUT -p tcp --dport ssh -j ACCEPT
iptables -A INPUT -p icmp --limit 3/sec \
            --limit-burst 5 -j ACCEPT
iptables -A INPUT -p tcp --dport auth -j REJECT
iptables -A INPUT -d 224.0.0.0/4 -p udp \
                  --dport 1234 -j ACCEPT
iptables -A INPUT -d SAP.MCAST.NET -p udp \
                  --dport 9875 -j ACCEPT
iptables -A INPUT -p igmp -j ACCEPT
#A teď to nejdůležitější !
iptables -P INPUT DROP
```

Jak uložit nastavení trvale?

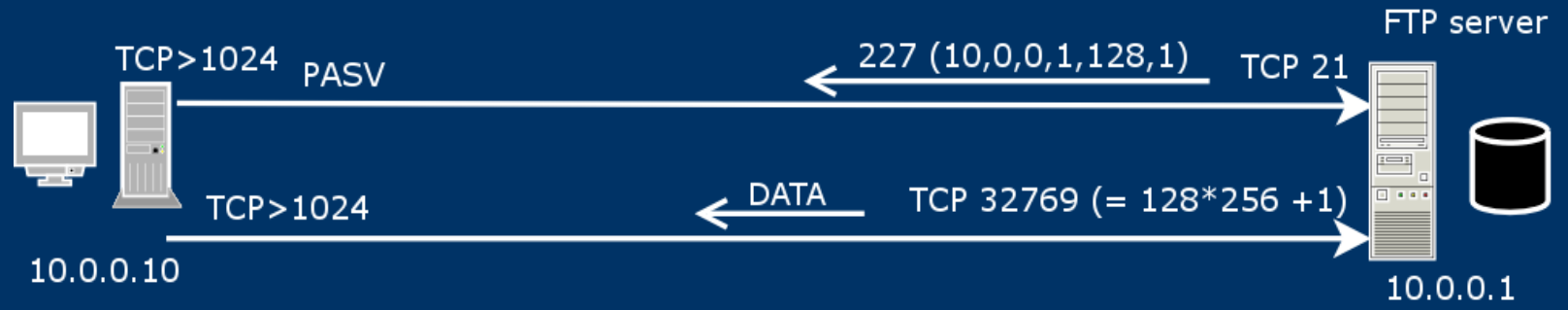
- Vytvořit ze sady pravidel initskript – nedoporučuje se používat jiné než číselné názvy
- Příkazem *iptables-save*:
  - Uloží na std. výstup (!) stav netfiltru ve formě sekvence argumentů příkazu *iptables*, kterými se do prázdného netfiltru současný stav replikuje
  - Tento výstup je také vhodný pro zjištění stavu netfilteru.
  - Stav netfilteru se obnoví příkazem *iptables-restore*
  - Distribuce to obvykle automatizují */etc/init.d/iptables save*

A co FTP?

- Aktivní režim:



- Pasivní režim:



Simulace FTP klienta

- Řídicí spojení používá protokol TELNET:

```
$ telnet ftpserver ftp
```

- Na datové spojení poslouží netcat – TCP/IP Swiss Army Knife

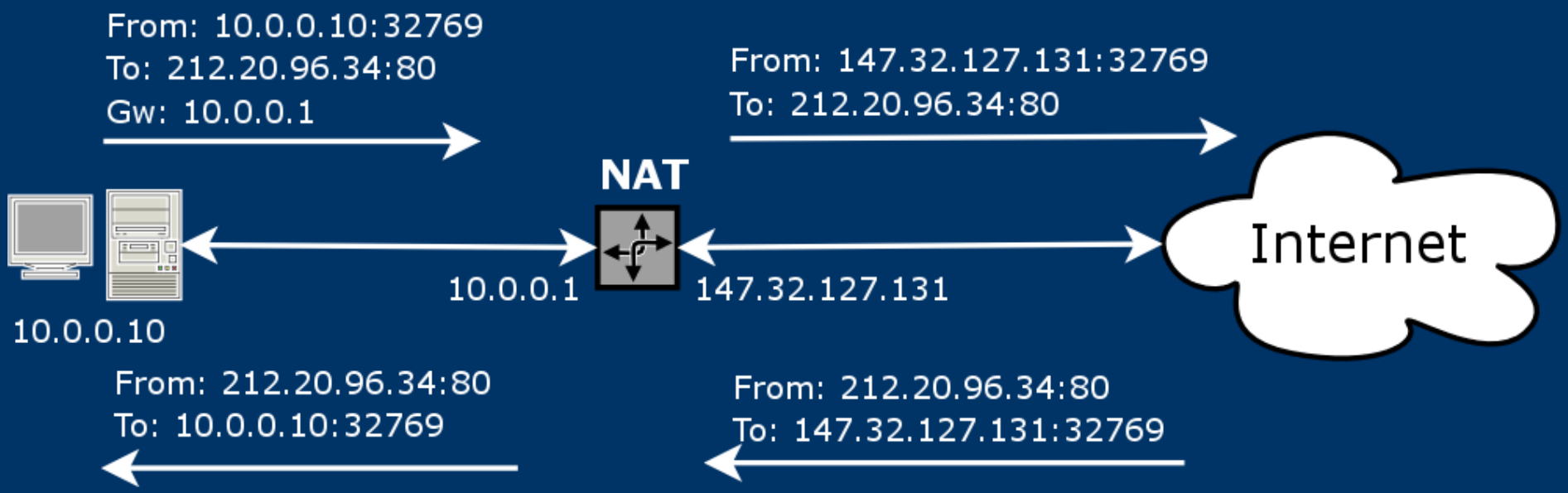
```
pro aktivní režim:
$ nc -l port

pro pasivní režim:
$ nc ftpserver port
```

FTP a firewall/NAT

- S uvedeným firewallem bez problému funguje pasivní FTP spojení z klienta za firewallem na server vně a naopak aktivní spojení z klienta vně na server uvnitř.
- Pro ostatní režimy je potřeba přidat do jádra modul *ip_conntrack_ftp*, který zajistí sledování řídicího spojení na portu 21 a vyhodnocování datových spojení jako RELATED.
- Pro NAT je ještě potřeba přidat modul *ip_nat_ftp*, který zajistí překlad adres v příkazech PORT a 227

NAT – překlad síťových adres



Rozjetí NATu v iptables

- Nastavení pravidla pro SNAT:

```
iptables -t nat -A POSTROUTING -s 10.0.0.0/24 \
    -d \! 10.0.0.0/24 -j SNAT -to-source 147.32.127.131
```

- Povolení forwardování paketů

```
echo 1 > /proc/sys/net/ipv4/ip_forward
```

- Port forwarding

```
iptables -t nat -A PREROUTING -p tcp --dport 10022 \
         -j DNAT --to-destination 10.0.0.10:22
```

Závěr

- Dále pomůže manuálová stránka iptables
- Na linuxu obvykle běží pouze služby, které chceme. Proto je firewall spíše specialitou, nikoli nutností.
- Pěkný návod o síťování v linuxu na http://www.pslib.cz/ke/manuals/linux/pripojeni/