Zjednodusene zaklady prace s IPTABLES

Jiri Kubina

jiri.kubina@osu.cz

Ver. 1.1

zari 2006

Obsah

Upozorneni: **Tento material si nedava za cil byt vycerpavajicim manualem. Jedna se o vyukovy material, zabyvajici se pouze zakladnimi principy. Podrobnosti ziskate prikazem man iptables nebo na www.netfilter.org**

1.Rozdeleni firewallu

- packet filter /paketovy filtr/
- statefull inspection firewall /stavovy firewall/
- application proxy gateway firewall /aplikacni proxy firewall/
- network address translation /preklad sitovych adres/
- hybrid firewall technologies /napriklad analyzatory paketu IDS,IPS/
- personal firewall

2.Co umi iptables ?

- packet filter /paketovy filtr/
- statefull inspection firewall /stavovy firewall/
- network address translation /preklad sitovych adres/
- personal firewall

- castecne /za pomoci specifickych modulu/ i

  application proxy gateway firewall /aplikacni proxy firewall/

3.Jak to funguje ?

V ruznych mistech pocitace, kterymi prochazi paket jsou umisteny ruzne druhy tabulek s ruznymi pravidly, ktera se aplikuji na prochazejici pakety v zavislosti kudy, kam, kdy a jaky paket prochazi.

Poznamka:

IP FORWARD

Routovani /preposilani/ paketu mezi jednotlivymi interfacy na linuxovem PC je nutno povolit.

Na vsech linuxech toho dosahneme budto rucne :

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

Pripadne systemovym nastavenim - CentOS /etc/sysctl.conf

```
net.ipv4.ip_forward = 1
```

4.Tables – Tabulky

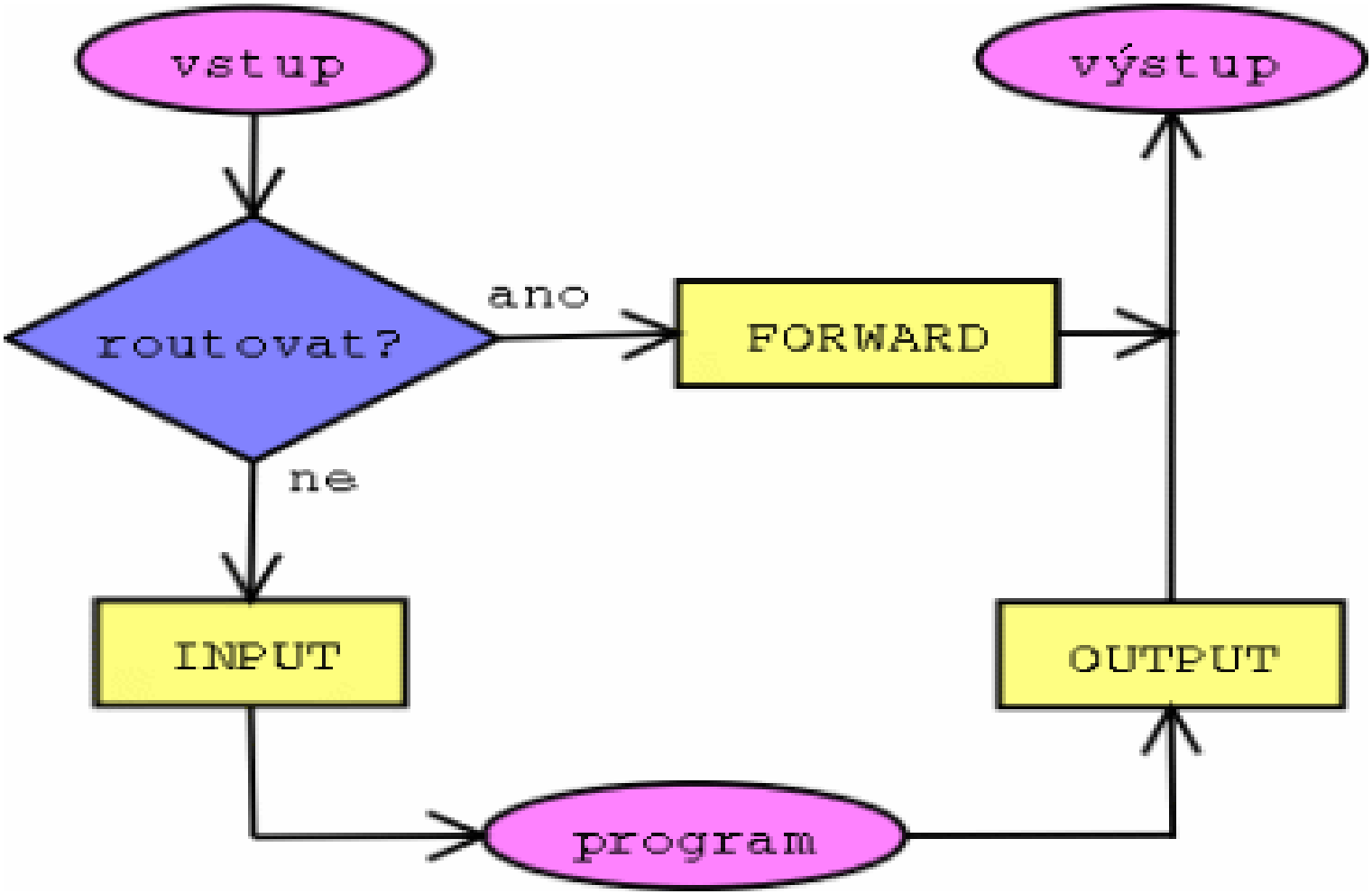
Nazvy tabulek /podle umisteni/ :

INPUT OUTPUT FORWARD PREROUTING POSTROUTING

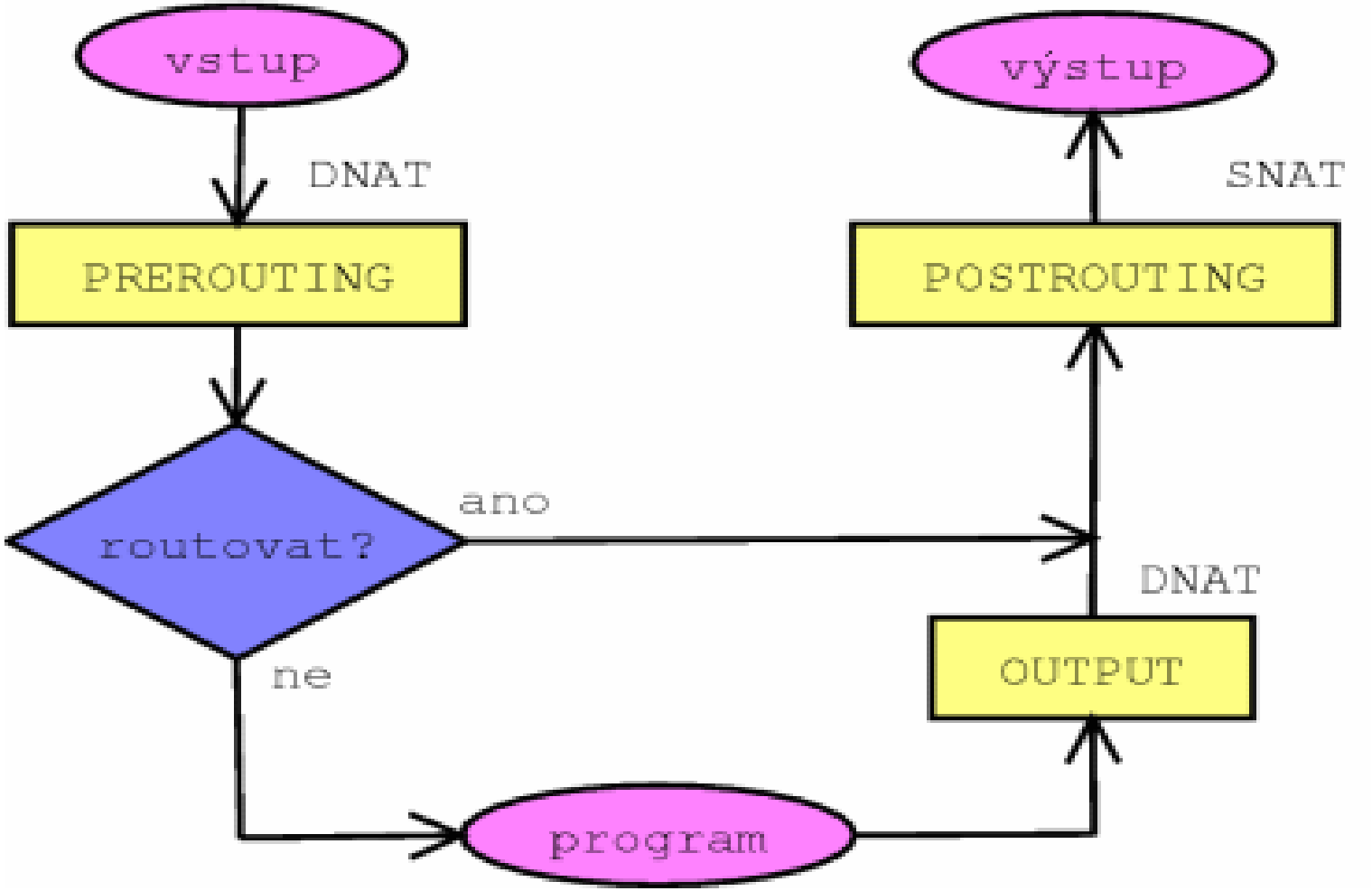
Druhy tabulek /podle funkce/ :

- FILTER - INPUT,OUTPUT,FORWARD
  slouzi k filtrovani /ACCEPT,DROP,REJECT/
- NAT - PREROUTING,OUTPUT,POSTROUTING slouzi k prekladu sitovych adres /SNAT,DNAT,MASQUERADE/
- MANGLE - PREROUTING,OUTPUT,INPUT,FORWARD,POSTROUTING
  slouzi ke zmene paketu /MARK,.../

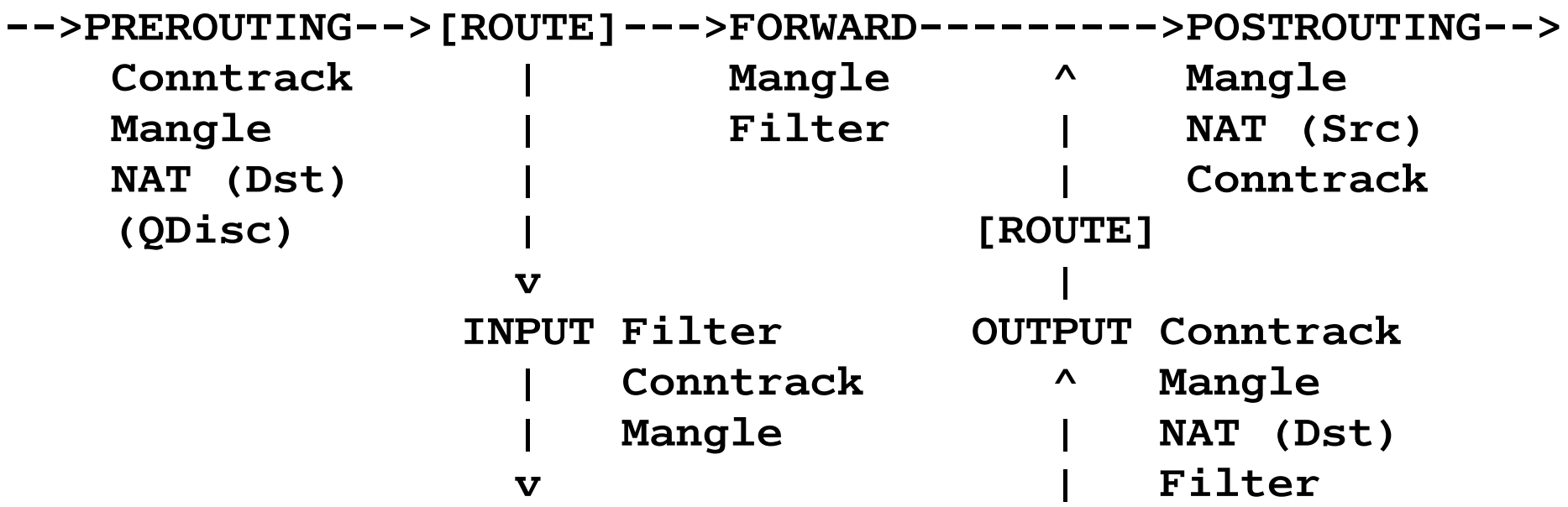
Tabulky FILTER



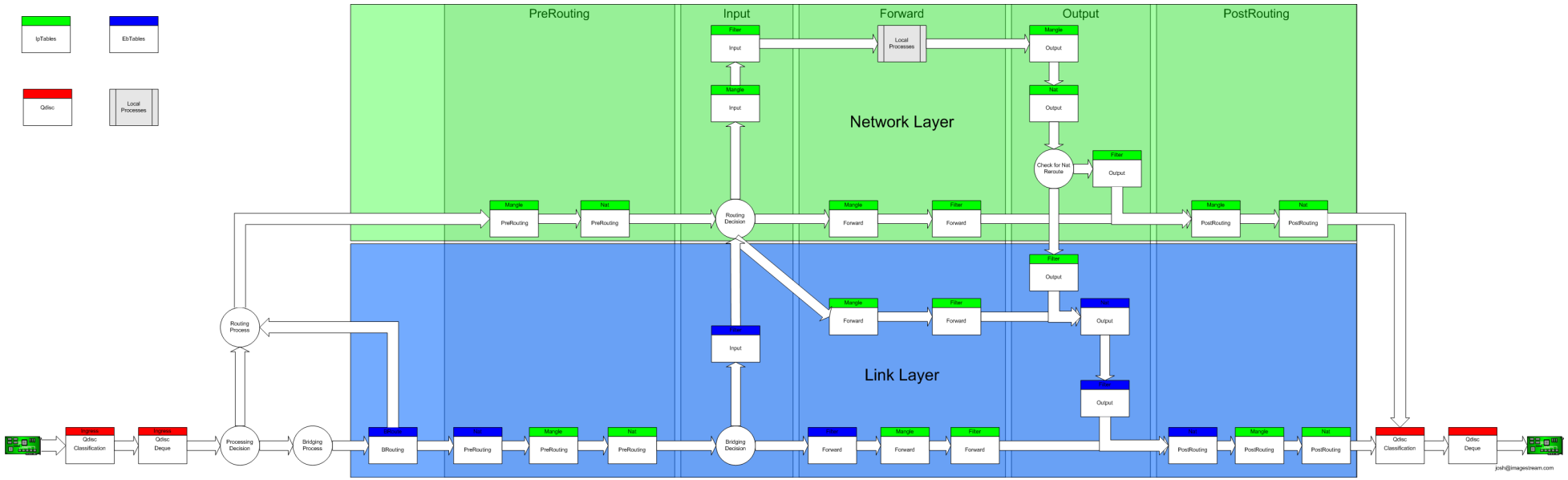
Tabluky NAT


vstup
DNAT
PREROUTING
routovat?
ano
ne
program
DNAT
OUTPUT
SNAT
POSTROUTING
výstup

Tables


-->PREROUTING-->[ROUTE]--->FORWARD---------->POSTROUTING-->
Conntrack
Mangle
NAT (Dst)
(QDisc)
Mangle
Filter
[ROUTE]
Mangle
NAT (Src)
Conntrack
INPUT Filter
Conntrack
Mangle
OUTPUT Conntrack
Mangle
NAT (Dst)
Filter

Global packet flow diagram



5.Targets /Targets extensions/ - Cile

- **ACCEPT** - pusti paket skrz tabulku
- **DROP** - zahodi paket
- **RETURN** - vrati paket do predchoziho retezce /ze ktereho se paket dostal do tohoto retezce/ nasledujicimu pravidlu
- **QUEUE** - pusti paket z kernelu do userspace /pro dalsi zpracovani - musi byt nakonfigurovano v kernelu/

Targets extensions /pouze nektere !/

- **DNAT** - Destination network address translation
- **LOG** - Logovani do syslogu
- **MARK** - Oznacovani paketu
- **MASQUERADE** Preklad adres na adresu odchoziho rozhrani
- **REJECT** - Odeslani chybove odpovedi na prijaty paket
- **SNAT** - Source network address translation

6.Commands – Prikazy

-A, --append Pridani noveho pravidla na konec retezce
-D, --delete Smaze pravidlo (bud ho zadate ve tvaru, v nemz jste ho pridavali, nebo pouzijete jeho cislo, to ziskate rozsirenou volbou --lin. Viz. dole).
-R, --replace Nahradi cislo pravidla jinym pravidlem
-I, --insert Vlozeni noveho pravidla na zacatek retezce
-L, --list Vypsani vsech pravidel v retezci. Pokud neni zadan retezec, vypisou se vsechny retezce + jejich pravidla
-F, --flush Vyprazdni vsechna pravidla v retezci (to same, jako kdybyste to delali po jednom)
-N, --new-chain Vytvorime si vlastni retez
-X, --delete-chain - Smazeme si vlastni retez (nejde smazat vychozi)
-P, --policy Vychozi politika retezce
-E, --rename-chain Prejmenovani vlastniho retezce
-Z, --zero Vynuluje vsechna pocitadla konkretniho retezce

7.Parameters - Parametry

-p, --protocol [!] protocol - [tcp|udp|icmp|gre|...]
-s, --source [!] address[/mask] – [10.0.0.2|10.0.0.0/24]
-d, --destination [!] address[/mask]– [10.0.0.2|10.0.0.0/24]
-j, --jump target - cil [ACCEPT|DROP|REJECT|RETURN|SNAT|...]
-i, --in-interface [!] name – [eth0|lo|br0|ppp0|eth1.3|..]
-o, --out-interface [!] name – [eth0|lo|br0|ppp0|eth1.3|..]
[!] -f, --fragment

8.Options – Volby

-v, --verbose
-n, --numeric
-x, --exact
--line-numbers
--modprobe=command

9.Match extensions – Vyhledavaci rozsireni /pouze nektere ! – moduly/

- **icmp**
  ```
  --icmp-type [!] typename
  ```

- **iprange**
  ```
  [!]--src-range ip-ip
  [!]--dst-range ip-ip
  ```

- **length**
  ```
  --length length[:length]
  ```

- **limit**
  ```
  --limit rate
  --limit-burst number
  ```

- **mac**
  ```
  --mac-source [!] address
  ```

- **mark**
  ```
  --mark value[/mask]
  ```
- **pkttype**
  ```
  --pkt-type [unicast|broadcast|multicast]
  ```
- **state**
  ```
  --state [INVALID|ESTABLISHED|NEW|RELATED]
  ```
- **tcp**
  ```
  --source-port [!] port[:port]
  --destination-port [!] port[:port]
  --tcp-flags [!] mask comp
  [!] --syn
  --tcp-option [!] number
  --mss value[:value]
  ```
- **udp**
  ```
  --source-port [!] port[:port]
  --destination-port [!] port[:port]
  ```

10. Priklad pouziti modulu recent

Jedna se o modul, ktery je schopen podle nastavenych kriterii vytvaret docasny seznam IP adres, ktere se pokouseji o spojeni na chraneny server a to bud na IP nebo i na konkretni port. Jedna se o jakysi greylist, ktery je casove promenny. Na zaklade vytvoreneho greylistu, je mozne s pakety dále pracovat /zahazovat,vracet,akceptovat/.Nize uvedeny postup je specificky pro sluzbu SSH, lze ho vsak aplikovat i na jine sluzby.

```
iptables -F INPUT
iptables -F recent_ssh
iptables -X recent_ssh

iptables -N recent_ssh

#povoleni pruchodu paketu z navazanych spojeni
iptables -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
#odsmerovani vsech paketu z portu 22 do retezce recent_ssh
iptables -A INPUT -p tcp -m tcp --dport 22 -j recent_ssh
#povoleni zadoucich adres natvrdo /vrati se zpet - nejsou ovlivneny modulem recent/
iptables -A recent_ssh -s 10.0.0.2 -j RETURN
#zapsani zdrojove adresy prichoziho paketu do tabulky recent_ssh
iptables -A recent_ssh -m recent --set --rsource --name recent_ssh
#kontrola na nepritomnost zdrojove adresy paketu v tabulce recent_ssh za poslednich 60
sekund vice nez 5x . pokud ne -J RETURN pokud ano jde nize
iptables -A recent_ssh -m recent ! --rcheck --hitcount 5 --seconds 60 --name \
recent_ssh --rsource -j RETURN
#kontrola a update na nepritomnost zdrojove adresy paketu v tabulce recent_ssh_log za
poslednich 60 sekund. pokud neni provede se update tabulky recent_ssh_log a -J LOG
pokud je jde nize
iptables -A recent_ssh -m recent ! --update --seconds 60 --name recent_ssh_log \
--rsource -j LOG --log-prefix "recent_ssh DROP: " --log-level 7
#update tabulky recent_ssh_log
iptables -A recent_ssh -m recent --set --name recent_ssh_log --rsource
#vraceni paketu s informaci o nedostupnosti adresy
iptables -A recent_ssh -j REJECT --reject-with icmp-admin-prohibited
```

11.Priklady pravidel - jednoduchy firewall

Skladba komplexniho pravidla iptables :

iptables [tabulka] [akce] [chain] [ip_část] [match] [target] [target_info]

iptables -t nat -A PREROUTING -i eth0 -p tcp -s 195.113.106.167 -d 195.113.106.168 --dport 5900 -j DNAT --to-destination 10.0.0.2:5900

Toto pravidlo rika ze u vsech TCP paketu, ktere prijdou z rozhrani eth0 s IP zdrojovou adresou 195.113.106.167, IP cilovou adresou 195.113.106.168 a cilovym portem 5900 se provede DNAT /Destination Network Address Translation/ na cilovou IP 10.0.0.2 cilovy port 5900 /prepise se v IP hlavicce cilova IP adresa/.

Jednoducha pravidla :

```
iptables -P INPUT DROP

iptables -A INPUT -i eth0 -p tcp -d 10.0.0.5 --dport 80 -j
ACCEPT

iptables -A INPUT -i eth0 -p udp -d 10.0.0.5 --dport 53 -j
ACCEPT

iptables -A INPUT -i eth0 -p icmp --icmp-type echo-request
-j ACCEPT

iptables -A INPUT -i eth0 -m state --state ESTABLISHED,
RELATED -j ACCEPT
```

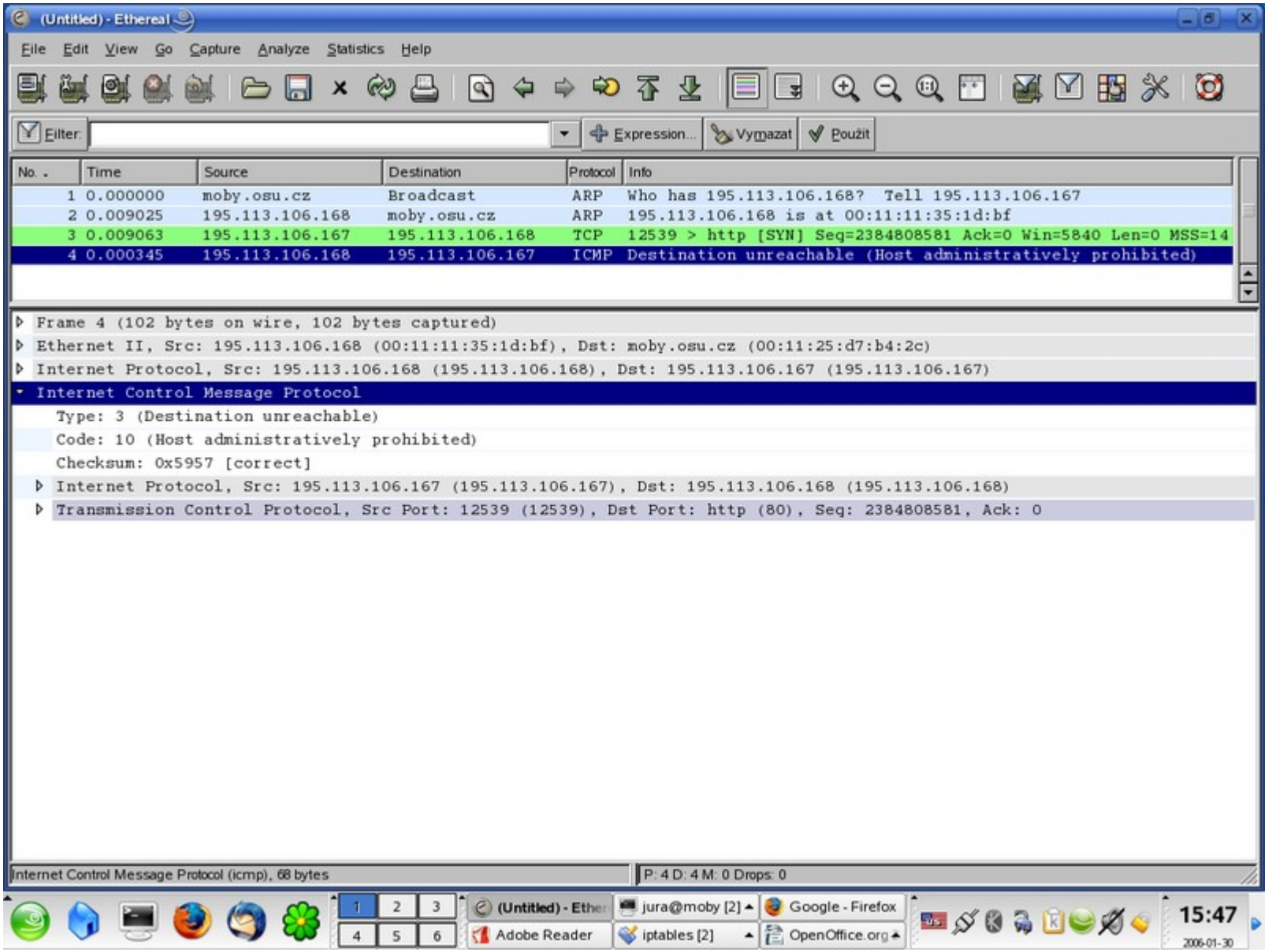
Jednoradkovy firewall :

```
iptables -A INPUT -m state --state NEW,INVALID -j DROP
```

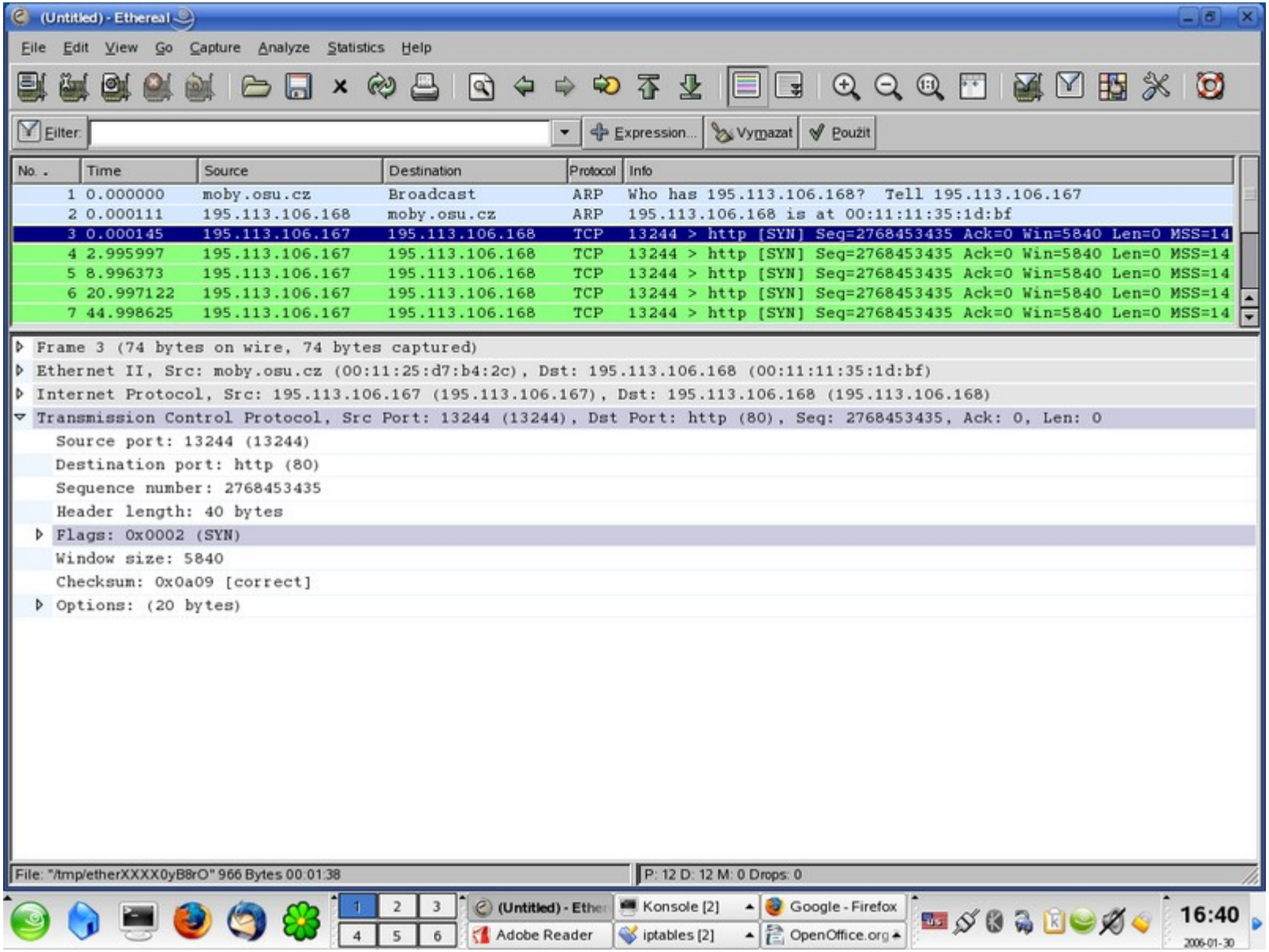
Priklad jednoducheho firewallu /host based/:

```
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -N CHECK_ICMP
iptables -N STOP_FLOODS
iptables -A INPUT -i lo -j ACCEPT
iptables -A INPUT -i eth0 -p icmp -j CHECK_ICMP
iptables -A INPUT -i eth0 -m state --state INVALID -j DROP
iptables -A INPUT -i eth0 -p tcp ! --syn -m state --state NEW -j DROP
iptables -A INPUT -i eth0 -p tcp --syn -j STOP_FLOODS
iptables -A INPUT -i eth0 -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A INPUT -i eth0 -p tcp --dport 113 -j REJECT --reject-with tcp-reset
iptables -A INPUT -m limit --limit 12/h -j LOG --log-prefix "INPUT drop: "
iptables -A INPUT -j REJECT --reject-with icmp-host-prohibited
iptables -A CHECK_ICMP -p icmp -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -A CHECK_ICMP -p icmp --icmp-type 0 -m length --length 28:84 -j ACCEPT
iptables -A CHECK_ICMP -p icmp --icmp-type 3 -m length --length 28:84 -j ACCEPT
iptables -A CHECK_ICMP -p icmp --icmp-type 8 -m length --length 28:84 -j ACCEPT
iptables -A CHECK_ICMP -p icmp --icmp-type 11 -m length --length 28:84 -j ACCEPT
iptables -A CHECK_ICMP -m limit --limit 12/h -j LOG --log-prefix "ICMP drop: "
iptables -A CHECK_ICMP -j DROP
iptables -A STOP_FLOODS -m limit --limit 1/s --limit-burst 5 -j RETURN
iptables -A STOP_FLOODS -j DROP
iptables -A OUTPUT -p tcp  -o eth0 -j ACCEPT
iptables -A OUTPUT -p udp  -o eth0 -j ACCEPT
iptables -A OUTPUT -p icmp -o eth0 -j ACCEPT
```

Chovani pravidla REJECT --reject-with icmp-host-prohibited



Chovani pravidla DROP



12.Pouzite zdroje a nastroje

Zdroje

- man iptables
- Oficialni web www.netfilter.org
- Clanky zabyvajici se problematikou iptables na serverech www.abclinuxu.cz www.root.cz
- V prezentaci jsou pouzity obrazky z clanku Stavime firewall http://www.root.cz/clanky/stavime-firewall-1/
- V prezentaci je pouzit obrazek z webu http://l7-filter.sourceforge.net/PacketFlow.png
- http://snowman.net/projects/ipt_recent/

Nastroje

- iptables
- ethereal

Dekuji za pozornost