Firewalls mit Iptables

Firewalls für den Linux Kernel 2.4

Was ist eine Firewall?

- Kontrolliert den Datenfluss zwischen dem internen Netz und dem Rest der Welt.

Es wird unterschieden:

- Statischer Packet-Filter
- Dynamischer Packet-Filter
- Applikationsgateway
- Hybridsysteme

TCP-IP Grundlagen

- TCP
- UDP
- Ports
- ICMP

Wichtige Begriffe

- NAT (Network Adress Translation)
- SNAT (Source NAT)
- DNAT (Destination NAT)
- Masquerading
- Redirect
- Port Forwarding
- Accounting

Compilieren des Kernels (Linux-Kernel 2.4.0-test10)

- Network Options
  - Network Packet Filtering (on)
  - IP Netfilter Configuration
    - <M> Connection Tracking
    - <M> FTP Protocol Support
    - <M> Iptables Support
    - <M> Limit Match Support
    - <M> Mac Adress Match Support
    - <M> Multiple Port Match Support
    - <M> Connection State Match Support
    - <M> Packet Filtering Support

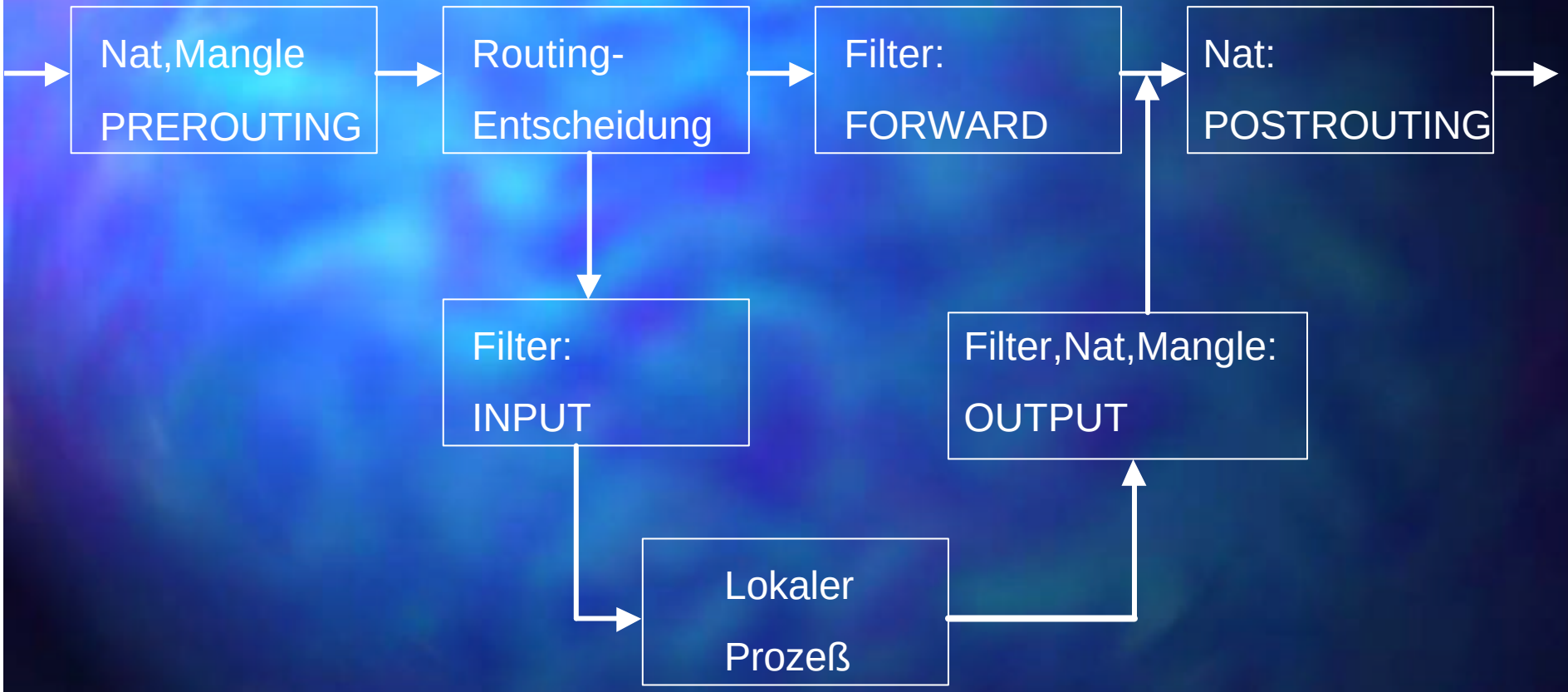
Compilieren des Kernels (Linux-Kernel 2.4.0-test10)

- Networking Options
  - IP Netfilter Configuration
    - <M> Reject target support
    - <M> Full NAT Support
    - <M> Redirect Target Support
    - <M> Log Target Support
- Iptables Software:
  - http://netfilter.kernelnotes.org

Netfilter und Iptables

- Netfilter: Allgemeine Filterstruktur im Kernel
- Iptables (IPv4 und IPv6)

Datenfluß durch den Kernel



Tabellen und Ketten (chains)

- Filter - INPUT,FORWARD,OUTPUT
- Nat - PREROUTING,OUTPUT POSTROUTING
- Mangle - PREROUTING,OUTPUT

Filter-Policies

- Accept
- Drop
- Queue (nicht weiter behandelt)

Wichtige Aktionen von Iptables

- Accept
- Drop
- Reject
- Log
- Return

Filter Rules – Sruktur

- Module laden
- Hilfsvariablen definieren
- Default Policy
- Ketten anlegen
- Filterregeln für die Ketten definieren

Erstes Beipiel

```
#!/bin/sh
modprobe ip_tables

iptables -A INPUT -j DROP
iptables –A FORWARD -j DROP
```

Zweites Beispiel

```
#!/bin/sh
modprobe ip_tables
iptables -F
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -P FORWARD DROP
```

Drittes Beispiel

```
modprobe ip_tables
modprobe ip_conntrack
modprobe ip_conntrack_ftp
modprobe ipt_state

iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -P FORWARD DROP
iptables -N block
iptables -A block -m state -state ESTABLISHED,RELATED -j ACCEPT
iptables -A INPUT -j block
iptables -A FORWARD -j block
```

Viertes Beispiel (Übersicht)

<Module>
<Variablen>
<Policy>
<Chain Def.>
<Blocking>
<Open Ports>
<Logging(1)>
<ICMP>
<Backdoor for Admin>
<Logging(2)>

Viertes Beispiel <module>

modprobe ip_tables
modprobe ip_conntrack
modprobe ip_conntrack_ftp
modprobe ipt_state
modprobe ipt_limit
modprobe ipt_mac
modprobe ipt_multiport
modprobe ipt_REJECT
modprobe ipt_LOG

Viertes Beispiel <Variablen>

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

IDEV=eth0 #World

IDEV=eth1 #Local Network

LIP0=129.13.30.1

LIP1=129.13.31.1

Viertes Beispiel <Policy>

```
iptables -F        # delete all rules and chains
iptables -P INPUT DROP
iptables -P OUTPUT ACCEPT
iptables -P FORWARD DROP
```

Viertes Beispiel <Chain Def.>

```
iptables -N block
iptables -A INPUT -j block
iptables -A FORWARD -j block

iptables -A block -m state --state ESTABLISHED,RELATED
    -j ACCEPT
```

Viertes Beispiel <Blocking>

iptables -A block -s m210 -j DROP

iptables -A block -s 129.77.13.0/255.255.255.0 -j REJECT

Viertes Beispiel <Open Ports>

```
# open PORT SSH/SMTP/HTTP/SMB/TALK
iptables -A block -p tcp --dport 22 -j ACCEPT
iptables -A block -p tcp --dport 25 -j ACCEPT
iptables -A block -p tcp --dport 80 -j ACCEPT
iptables -A block -p udp --dport 137:139 -j ACCEPT
iptables -A block -p udp --dport 517:518 -j ACCEPT
# open Ports for localhost
iptables -A block -s servername -j ACCEPT
iptables -A block -s localhost -j ACCEPT
```

Viertes Beispiel <Logging(1)>

```
# Logging von icmp und Verbindungsversuchen

iptables -A block -p icmp -j LOG --log-level notice --log-
    prefix “ICMP “ -m limit --limit 1/s

#Optional alle restlichen Verbindungsversuche loggen

iptables -A block -p tcp --syn -j LOG --log-level notice --
    log-prefix “SYN “ -m limit --limit 1/s
```

Viertes Beispiel <ICMP>

iptables -A block -p icmp --icmp-type echo-request -m limit --limit 1/s –j ACCEPT

iptables -A block -p icmp --icmp-type \! echo-request –j ACCEPT

Viertes Beispiel <Backdoor>

```
iptables -A block -m mac --mac-source
    08:00:20:03:5F:DA -s m406b -i $IDEV1 -j ACCEPT
```

Viertes Beispiel <Logging(2)>

```
# Not Logging Broadcast crap
iptables -A block -p udp -d 255.255.255.255 -j RETURN
iptables -A block -p udp -d 129.13.30.255 -j RETURN
iptables -A block -p udp -d 129.13.31.255 -j RETURN
# SYN Packets already logged
iptables -A block -p tcp --syn -j RETURN
# LOG all not accepted UDP-Packets
iptables -A block -p udp -j LOG --log-level notice --log-prefix “UDP “
    -m limit --limit 1/s
# Should not happen
iptables -A block -p tcp -j LOG --log-level notice --log-prefix “TCP “
    -m limit --limit 1/s
```