!!! ACHTUNG - evtl. veraltet - ACHTUNG !!!

Diese Seite wurde zuletzt am 8. Juli 2014 um 17:02 Uhr geändert.

Grundsätzlich sollte man alles in ein Shell-Skript packen.

Darüber hinaus dann natürlich die Umgebungsvariable „PATH“ anpassen, das ausführbare Binary nebst Pfad und die Netzwerkschnittstellen in eine Variable schreiben, z. B. so:

```
#!/bin/bash
export PATH="$PATH:/sbin"
IPTABLES="/sbin/iptables"
RED="eth0"
GREEN="dummy0"
[[...]]
```

Bevor die eigentlichen Regeln dran sind, können noch ein paar grundlegende Einstellungen gemacht werden, z. B. so:

```
[[...]]
if [[|-e /proc/sys/net/ipv4/conf/all/rp_filter ]]; then
  for f in /proc/sys/net/ipv4/conf/*/rp_filter; do
    echo 1 > $f
  done
fi
echo 0 > /proc/sys/net/ipv4/tcp_ecn
modprobe ip_conntrack
modprobe ip_conntrack_ftp
modprobe ip_nat_ftp
[[...]]
```

Da man die Regeln bestimmt nicht nur aktivieren, sondern auch deaktivieren möchte, packt man sich das ganze am Besten gleich per „case“ in die Alternativen „start“, „stop“, „restart“ und „reset“.

Start könnte z. B. so aussehen:

```
# routing (off) + DROP all + cleanup
echo 0 > /proc/sys/net/ipv4/ip_forward
$IPTABLES -P INPUT DROP
$IPTABLES -P OUTPUT DROP
$IPTABLES -P FORWARD DROP
$IPTABLES -t nat -F
$IPTABLES -t nat -X
$IPTABLES -F
$IPTABLES -X
# input
$IPTABLES -N RED-INPUT
$IPTABLES -A INPUT -i lo -j ACCEPT
$IPTABLES -A INPUT -i $GREEN -j ACCEPT
$IPTABLES -A INPUT -i $RED -j RED-INPUT
$IPTABLES -A RED-INPUT -p tcp --dport 22 -j ACCEPT
```

```
 # output
 $IPTABLES -N RED-OUTPUT
 $IPTABLES -A OUTPUT -o lo -j ACCEPT
 $IPTABLES -A OUTPUT -o $GREEN -j ACCEPT
 $IPTABLES -A OUTPUT -o $RED -j RED-OUTPUT
 $IPTABLES -A RED-OUTPUT -p tcp --dport 21 -j ACCEPT
 $IPTABLES -A RED-OUTPUT -p tcp --dport 53 -j ACCEPT
 $IPTABLES -A RED-OUTPUT -p udp --dport 53 -j ACCEPT
 $IPTABLES -A RED-OUTPUT -p tcp --dport 80 -j ACCEPT
 $IPTABLES -A RED-OUTPUT -p tcp --dport 123 -j ACCEPT
 $IPTABLES -A RED-OUTPUT -p udp --dport 123 -j ACCEPT
 $IPTABLES -A RED-OUTPUT -p tcp --dport 443 -j ACCEPT
 $IPTABLES -A RED-OUTPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
 $IPTABLES -A RED-OUTPUT -j REJECT
 # forward
 $IPTABLES -N FORWARD-GREEN
 $IPTABLES -N FORWARD-RED
 $IPTABLES -A FORWARD -i $GREEN -j FORWARD-GREEN
 $IPTABLES -A FORWARD -i $RED -j FORWARD-RED
 $IPTABLES -A FORWARD -m state --state ESTABLISHED,RELATED -j ACCEPT
 $IPTABLES -A FORWARD -j DROP
 $IPTABLES -A FORWARD-GREEN -o $GREEN -j ACCEPT
 $IPTABLES -A FORWARD-GREEN -o $RED -j ACCEPT
 $IPTABLES -A FORWARD-GREEN -j REJECT
 $IPTABLES -A FORWARD-RED -j DROP
 # dnat
 $IPTABLES -t nat -A PREROUTING -i $RED -p tcp --dport 25 -j DNAT --to-
destination 192.168.0.25:25
 # snat
 $IPTABLES -t nat -A POSTROUTING -s 192.168.0.0/255.255.255.0 -d !
192.168.0.0/255.255.255.0 -j SNAT --to-source <IP_ADRESSE_DES_HOSTS>
 # routing (on)
 echo 1 > /proc/sys/net/ipv4/ip_forward
 # tws => 64k (max)
 echo 0 > /proc/sys/net/ipv4/tcp_window_scaling
```

Stop könnte z. B. so aussehen:

```
 # ACCEPT all + cleanup
 $IPTABLES -P INPUT ACCEPT
 $IPTABLES -P OUTPUT ACCEPT
 $IPTABLES -P FORWARD ACCEPT
 $IPTABLES -t nat -F
 $IPTABLES -t nat -X
 $IPTABLES -F
 $IPTABLES -X
```

Ein Restart ruft nacheinander „stop“ und „start“ auf.

Um die Counter (hilfreich zum auffinden von fehlenden Regeln per „iptables -n -L -v“) resetten zu können, könnte „reset“ z. B. so aussehen:

```
# reset counters
$IPTABLES -t nat -Z
$IPTABLES -Z
```

Hier und da fehlen viell. noch ein paar Informationen, aber für den Anfang sollte es das hier erstmal tun.