iptables – Die Firewall Netfilter

- Kurzakte –

iptables als Konfigurationswerkzeug für Netfilter		
Syntax: **iptables Tabelle Kettenoperation Regelkette Auswahlkriterien Aktion**		
Filtermöglichkeiten		
Sender IP-Adresse	• Host- oder Netzadressen	
Empfänger IP-Adresse	• Host- oder Netzadressen	
Sender-Port	• Client: unprivilegiert im Bereich 1024 bis 65535; Server: 1 bis 1023	
Empfänger-Port	• Client: unprivilegiert im Bereich 1024 bis 65535; Server: 1 bis 1023	
Protokolle	• TCP, UDP, ICMP	
SYN- und ACK-Flag	• identifiziert Verbindungsstatus	
Netzwerkschnittstelle	• eth0, eth1, pp0, ippp0, ...	
Richtung der Daten	• in, out	
Tabellen		
Auswahl mit: **iptables -t Tabelle** (Weglassen des Tabellennamens bewirkt die Nutzung von **filter**)		
filter	• Standardtabelle für allgemeine Firewallregeln des lokalen Rechners • Regelketten: INPUT, FORWARD und OUTPUT	
nat	• für IP-Masquerading oder Network Address Translation (NAT) • ändert die Absender- oder Empfängeradressen der Datenpakete • durchgeroutete Datenpakete passieren diese Tabelle nur einmal. • Die Regeln werden nur auf das erste Paket eines Datenstroms angewandt. Darf das Paket passieren werden alle weiteren Pakete dieses Datenstroms automatisch maskiert. • Regelketten: PREROUTING, OUTPUT und POSTROUTING.	
mangle	• Verändern einzelner Elemente des Headers (z. B. TTL, TOS oder MARK) • Regelketten: PREROUTING und OUTPUT	
Kettenoperationen		
------ Operationen betreffen die ganze Kette ------		
-P Regelkette Policy	• Die angegebene Policy wird zur Grundeinstellung der genannten Standard-Regelkette • möglich sind DROP oder ACCEPT	
-F Regelkette	• Alle bestehenden Regeln aller oder der angegebenen Kette werden gelöscht. Die Grundeinstellung (Policy) bleibt jedoch erhalten. • Wird keine Regelkette angegeben, so werden alle Regeln aller Regelketten gelöscht.	
-N Regelkette	• Erstellen einer neuen Kette (Name in Kleinbuchstaben empfohlen, max. 31 Zeichen)	
-E Regelkette	• Umbenennen einer bestehenden Kette	
-Z Regelkette	• Die Byte- und Paketzähler einer Kette auf Null setzen	
-X Regelkette	• Löschen einer oder aller leeren benutzerdefinierten Regelketten	
-L Regelkette	• Alle Regeln einer oder aller Ketten anzeigen • die Option –v sorgt für eine ausführlichere Ansicht mit Schnittstellen und Zählern • die Option –n sorgt für die numerische Ausgabe von Rechneradressen und Portnummern	
------ Operationen betreffen nur einzelne Regeln innerhalb einer Kette ------		
-A Regelkette ...	• Eine einzelne Regel wird ans Ende der angegebenen Regelkette angehängt.	
-I Regelkette ...	• Eine einzelne Regel wird an den Anfang der angegebenen Regelkette eingefügt.	
-R Regelkette …	• Eine einzelne Regel durch eine andere in der angegebenen Regelkette ersetzen	
-D Regelkette …	• Eine einzelne Regel aus der angegebenen Regelkette löschen	
Standard-Regelketten für die Tabellen (chains)		
OUTPUT	• betrifft alle selbst generierten Pakete, die den Rechner verlassen	Tabellen: filter, nat, mangle
INPUT	• betrifft alle Pakete, die der Rechner empfängt	Tabellen: filter
FORWARD	• betrifft alle Pakete, die der Rechner routen soll	Tabellen: filter
PREROUTING	• alle hereinkommenden Pakete werden verändert (DNAT)	Tabellen: nat, mangle
POSTROUTING	• alle ausgehenden Pakete werden verändert (SNAT)	Tabellen: nat
• es lassen sich weitere eigene Ketten definieren und auch wieder löschen (z. B. für verschiedene Netze)		

Auswahlkriterien für Paketauswahl (matches)	
die einzelnen Prüfkriterien werden und-verknüpft	
-i [!] Interface	• Die Input-Netzwerkschnittstelle, für die die Regel gilt. (INPUT, FORWARD) • eth+ steht für alle Ethernet-Devices (nicht eth*)
-o [!] Interface	• Die Output-Netzwerkschnittstelle, für die die Regel gilt. (OUTPUT, FORWARD)
-s [!] IP-Adresse [/Maske]	• Absenderadresse (Source) des Paketes und optional die Subnet-Maske
-d [!] IP-Adresse [/Maske]	• Empfängeradresse (Destination) des Paketes und optional die Subnet-Maske
-p [!] Protokoll	• das vom Paket verwendete Protokoll (tcp, udp, icmp, all) • ist Option –p nicht definiert, gilt "all" • ist Option –p ! udp definiert, gilt "alles, außer UDP" • die Angabe des Protokolls lädt automatisch die match-Erweiterungen dafür
--syn	• Das SYN-Flag einer TCP-Nachricht muss gesetzt, kein ACK • gilt für das erste Paket eines Verbindungsaufbaues, gesendet vom Client. (nur tcp)
[!] --syn	• Das ACK-Flag einer TCP-Nachricht muß gesetzt sein. • Das Paket ist entweder der zweite Teil des Verbindungsaufbaues oder es ist ein Teil einer bestehenden Verbindung. • entspricht der Langform --tcp-flags SYN,RST,ACK SYN • Ist weder --syn noch ! --syn gesetzt, werden die TCP-Flags nicht überprüft. (tcp)
--source-port [!] [port] oder --sport …	• Prüfen des Quellports (Quellportbereichs) vom Paket (tcp, udp)
--destination-port [port] oder --dport …	• Prüfen des Zielports (Quellportbereichs) vom Paket (tcp, udp)
einige modulare Erweiterungen für die matches	
-m length --length	• Überwachung der Mindestgröße von Paketen • Größenangabe in Byte als Einzelwert oder Bereich von:bis
-m limit --limit	• zeitliche Begrenzung der von einer Regel erfassten Pakete (gegen DOS-Attacken)
-m multiport –p tcp|udp	• erlaubt die Angabe von bis zu 15 Quell- oder Zielports • nur in Verbindung mit -p tcp oder -p udp (-p tcp -m multiport --dports 80, 443)
-m state --state paketstatus	• Stateful Firewalling prüft den Zustand aller Verbindungen (nicht auf Paketebene) • Nutzung der Module state, ip_conntrack, ip_conntrack_ftp • ist deutlich effizienter als die Prüfung jedes Einzelpakets (z.B. mit --syn) • mögliche Paket-Stati: NEW - das Paket eröffnet eine neue Verbindung ESTABLISHED - Paket gehört zu einer bestehenden Verbindung INVALID - Paket gehört zu keiner bekannten Verbindung RELATED - Paket steht in Beziehung zu einer bereits aufgebauten Verbindung (FTP, ICMP-Fehlermeldungen) • Sichten mit "cat /proc/net/ip_conntrack"
Aktionen (targets)	
ACCEPT	• Paket wird akzeptiert
DROP	• Paket wird verworfen, ohne dem Absender eine Rückmeldung zu geben
QUEUE	• Einreihen des Pakets in eine Warteschlange für den angegebenen Benutzerprozess • also die Weiterleitung an ein Programm, falls der Kernel dies unterstützt
RETURN	• Verlassen der Chain durch Sprung an deren Ende
REJECT	• wie DROP, aber sendet Fehlermeldung an den Absender (icmp: port unreachable) • andere ICMP-Meldungen mit " –j REJECT --reject-with icmp-host-unreachable "
LOG	• Mitloggen der definierten Pakete ohne deren Beeinflussung beim Durchlauf • Protokollierung erfolgt über das Kernel-Log via printk() – üblicherweise syslogd
MARK, TOS	• speziell für die Tabelle MANGLE
REDIRECT	• lokal generierte Pakete werden auf die Zieladresse 127.0.0.1 gesetzt
MASQUERADE	• ändert Quelladresse auf die dynamisch zugewiesene IP-Adresse bei dial-up-Verbindungen (nur für POSTROUTING von NAT)
SNAT	• die Quelladresse soll modifiziert werden (nur für POSTROUTING von NAT)
DNAT	• die Zieladresse soll modifiziert werden (nur PREROUTING und OUTPUT von NAT)
eigene Kette	• Sprungziel kann auch eine selbst erstellte Regelkette sein
wichtige Optionen für das Target LOG	
--log-level	• legt Priorität für syslogd fest (numerisch oder ausgeschrieben)
--log-prefix	• Erläuterung zur besseren Analyse dieser Logmeldung (max. 27 Zeichen)

Darstellungsart von Ports und Adressen	
192.168.10.12/24	IP-Adressen können mit einer Maske versehen werden
192.168.10.12/32	Adresse muß exakt übereinstimmen
192.168.10.12/0	kein Bit muß übereinstimmen (alle Adressen möglich)
any/0	w. o.
1024:65535 oder 1024:	ganzer Bereich von gültigen Portnummern (Startport:Endport)
Formulierung von Regeln	

Für jede Art von Datenverbindung müssen mindestens zwei Regeln formuliert werden (falls nicht -m state):
• eine für die input-chain und eine für die output-chain:
`iptables -t Tabelle -A Regelkette -i in-Interface -o out-I. -p Protokoll \`
`-s Absenderadresse --sport Absenderport \`
`-d Empfängeradresse --dport Empfängerport -j Policy`	

Sichern der im Kernel geladenen Filterregeln	
iptables-save	• die derzeit gesetzten Regeln inklusive Policies und Zählerständen auf der Standardausgabe ausgeben • alternativ Umleitung in eine Datei
iptables-restore	• Einlesen des Inhalts einer erstellten Backup-Datei
logrotate der mitgeschnittenen Netfilter-Kernelaktionen	
/etc/syslog.conf	• Zeile einfügen: `kern.info /var/log/kern.info` • Achtung, Leerraum nur mit Tabs auffüllen! • /etc/init.d/syslogd restart zum Neueinlesen der Konfigdatei

/etc/logrotate.d/kerninfo erstellen mit folgendem Inhalt:	`/var/log/kern.info {`
	`compress`
	`rotate 4`
	`missingok`
	`size +4096K`
	`create 644 root root`
	`}`

/etc/logrotate	Eintrag folgender Zeile: `/var/log/kern.info +8192k 640 root.root`
Dokumentation	
man: iptables(8)	

einfache Regeln zum Zugriff auf einen beliebigen Webserver:
Je eine Regel für ein- und ausgehende Pakete. Also zwei Regeln für eine Verbindung.

```
iptables -A OUTPUT –o eth0 -p tcp -s $OWN_IP --sport 1024: \
          -d any/0 --dport 80 -j ACCEPT
iptables -A INPUT -i eth0 -p tcp ! --syn -s any/0 --sport 80 \
          -d $OWN_IP --dport 1024: -j ACCEPT
```

Regeln mit connection tracking mit dem state-Modul:
Man erklärt einfach Pakete, die Teil einer existierenden Verbindung sind (ESTABLISHED) oder anderweitig zu ihr gehören (RELATED) für in Ordnung.

```
iptables -t filter -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
iptables -t filter -A OUTPUT -m state --state ESTABLISHED,RELATED -j ACCEPT
```

Jetzt muss man sich nur noch um das Aufbauen von Verbindungen (Status NEW) kümmern. Man benötigt nur noch eine Regel pro Verbindung für das erste Paket. Dies halbiert die Anzahl der erforderlichen Regeln:

```
iptables -A OUTPUT –o eth0 –p tcp -s $OWN_IP --sport 1024: \
          -d any/0 --dport 80 -m state --state NEW -j ACCEPT
```

Definition einer benutzerdefinierten Logging-Kette:

```
iptables -N logdrop
iptables -A -t logdrop -j LOG
iptables -A -t logdrop -j DROP
```

Anschließend können Sie überall da, wo Sie bisher »-j DROP« gesagt haben, »-j logdrop« verwenden.

```
linux:~ # iptables --help
iptables v1.2.8

Usage: iptables -[AD] chain rule-specification [options]
       iptables -[RI] chain rulenum rule-specification [options]
       iptables -D chain rulenum [options]
       iptables -[LFZ] [chain] [options]
       iptables -[NX] chain
       iptables -E old-chain-name new-chain-name
       iptables -P chain target [options]
       iptables -h (print this help information)

Commands:
Either long or short options are allowed.
  --append  -A chain            Append to chain
  --delete  -D chain            Delete matching rule from chain
  --delete  -D chain rulenum    Delete rule rulenum (1 = first) from chain
  --insert  -I chain [rulenum]  Insert in chain as rulenum (default 1=first)
  --replace -R chain rulenum
                                Replace rule rulenum (1 = first) in chain
  --list    -L [chain]          List the rules in a chain or all chains
  --flush   -F [chain]          Delete all rules in  chain or all chains
  --zero    -Z [chain]          Zero counters in chain or all chains
  --new     -N chain            Create a new user-defined chain
  --delete-chain
            -X [chain]          Delete a user-defined chain
  --policy  -P chain target     Change policy on chain to target
  --rename-chain
            -E old-chain new-chain
                                Change chain name, (moving any references)
Options:
  --proto       -p [!] proto    protocol: by number or name, eg. `tcp'
  --source      -s [!] address[/mask]
                                source specification
  --destination -d [!] address[/mask]
                                destination specification
  --in-interface -i [!] input name[+]
                                network interface name ([+] for wildcard)
  --jump        -j target
                                target for rule (may load target extension)
  --match       -m match
                                extended match (may load extension)
  --numeric     -n              numeric output of addresses and ports
  --out-interface -o [!] output name[+]
                                network interface name ([+] for wildcard)
  --table       -t table        table to manipulate (default: `filter')
  --verbose     -v              verbose mode
  --line-numbers                print line numbers when listing
  --exact       -x              expand numbers (display exact values)
[!] --fragment  -f              match second or further fragments only
  --modprobe=<command>          try to insert modules using this command
  --set-counters PKTS BYTES     set the counter during insert/append
[!] --version   -V              print package version.
```

grafische Tools

Konfiguration:

Firewall Builder (libfwbuilder + fwbuilder)
Guarddog – beherrscht kein Masquerading (make install prefix=/opt/kde3/)
Firestarter
Shoreline Firewall – mit eigenem Webmin-Plugin (bei Debian mitgeliefert)

Logfile-Analysetools:

Iptables Log Analyzer, (interessant durch Datenbankunterstützung)
Wallfire Wflogs (mächtige Abfragefilter)
Fwlogwatch (interssanter Echtzeitmodus)