Ip tables

Wir bastel'n uns ne Firewall

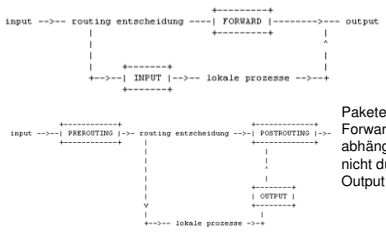
Paketfilter

- Iptables ist der eingebaute Paketfilter bei Linux ab Kernel 2.x.
- Ziel ist es, den Linux Server so zu konfigurieren, das er Verbindungen die an ihn gerichtet sind, über ihn an andere Server gerichtet sind oder die von ihm nach aussen gehen akzeptiert oder verwirft.
- Grundregel: „Die 1. Passende Regel gewinnt"

Tabellen

- Standardmäßig verfügt iptables über 3 voreingestellte Tabellen
  - INPUT
    - eingehende Pakete
  - FORWARD
    - weitergeleitete Pakete
  - OUTPUT
    - ausgehende Pakete

Welche Chains werden durchlaufen ?



Pakete die auf Forward gesetzt sind, abhängig von -d gehen nicht durch die Input und Output Chain

Regeln zum Anlegen und Löschen von Regeln einer Chain

- -A | --append <Chain>
  - -A Input
- -I | --insert <Chain>
- -D | --delete <Chain>

Vergleichsoperationen

- -i | --interface [!] <interface>
  - auf welches Netzwerkinterface bezieht sich die nachfolgende Regel
- -p | --protocol [!]
  - Gibt das IP Protokoll an, auf das sich die Regel beziehen soll
- -s | --source [!] <Adresse>
  - Gibt eine Ip Adresse oder ein Netzwerk als Empfängeradresse im Header an

Vergleichsoperationen

- -d | --destination | --dst [!] <Adresse>
  - Gibt eine IP Adresse als Empfängeradresse im Header an
- -j | --jump <Target>
  - Gibt an, was mit Paketen geschehen soll, die zu der gerade formulierten Regel passen. Voreingestellte Targets sind ACCEPT oder DROP. Erweiterung sind REJECT und LOG

Vergleichoperationen (tcp)

- --source-port | --sport [!] <Port>
  - Angabe des Absender-Ports
- --destination-port | --dport [!] <Port>
  - Angabe des Empfänger-Ports
- --tcp-flags [!] <Flag>

Erweiterungen für TARGET log

- --log-level <Syslog-LogLevel>
  - Entsprechen den Log Levels in der Syslog.conf
- --log-prefix <„Beschreibung“>
  - Als Beschreibung kann - Anführungszeichen nicht vergessen - ein beliebiger String angegeben werden der mit ausgegeben wird, wenn diese Regel eine Protokollmeldung auslöst

Erweiterungen für TARGET REJECT -j REJECT

- --reject-with <ICMP -3-Subtyp>
  - Voreinstellung icmp-port-unreachable wenn die Annahme verweigert wird. Es ist möglich stattdessen einen beliebigen Subtyp vom Typ 3 anzugeben
- --reject-with tcp-reset
- --reject-with echo-reply
  - Bei abgelehnten Ping-Echo-Requests kann als Fehlermeldung eine simulierte Antwort zurückgegeben werden. Die FW erzeugt eine Antwort auf das Ping, ohne die Anfrage an den eigentlichen Host weiterzuleiten

Verbindungszustände

- -m | --match-state <Zustand>
  - trifft zu, wenn sich die Verbindung in einem der angegebenen Zuständer befindet. Erlaubte Zuständer sind NEW, ESTABLISHED und RELATED

BSP

- Lokaler DNS Server. Leitet Anfragen an einen Server im Netz weiter und hat einen Zwischenspeicher für die Ergebnisse. Für die Weiterleitung der Anfragen an den Server im Netz benimmt er sich aber wie ein echter Server, d.h. der DNS Datenaustausch benutzt den Port 53 auf beiden beteiligten Rechnern. Die folgenden Regeln erlauben nun neue Anfragen (NEW), ankommende Antworten (ESTABLISHED) sowie ICMP Fehlermeldungen, die im Zustand mit der Anfrage stehen (RELATED).

Regeln hierfür

- iptables -A INPUT -m state --state ESTABLISHED, RELATED -j ACCEPT
- iptables -A OUTPUT --out-interface eth0 -p UDP -s <IPADR> --source-port 53 -d $NAME_SERVER --destination-port 53 -m state --state NEW, RELATED -j ACCEPT

Multiport Erweiterung

- Bei Multiport kann sich eine einzige Regel auf bis zu 15 Ports beziehen. Innerhalb der Liste sind Leerzeichen nicht erlaubt, d.h. zwischen den Kommas und den Ports darf kein Abstand bleiben. Der Befehl -m multiport muss unmittelbar nach der Protokollangabe -p <Protokoll> stehen.

Syntax

- --source-port <Port>[,Port] Angabe der Absenderports
- --destination-port <Port>[,Port] Angabe der Empfängerports

Bsp.

- Folgende Regel sperrt ankommende Pakete an die UDP Ports für NFS und lockd
  - iptables -A INPUT -i eth0 -p udp -m multiport --destination-port 2049,4045 -j DROP
- Sperren von abgehenden Verbindungsanfragen an hohe Ports die zu den TCP Diensten NFS, socks und squid gehören
  - iptables -A OUTPUT -o eth0 -p tcp -m multiport --destination-port 2049,1080,3128 --syn -j REJECT

Welche Regel findet Anwendung ?

- „Die erste Regel gewinnt „
  - Wie beim Doppelkopf - Ausnahme Herz 10

Filterregeln hier Printerport 515

- iptables -A INPUT -p tcp --dport 515 -j DROP
- -A die Regel wird an die Chain INPUT angehängt (append)
- -p Welches Protokoll ? - (tcp)
- -dport Destination Port - Wo soll's denn hingehen ?
- -j Jump - Zu welchem Target soll's denn gehen - Drop

Loggen von gedropten Paketen

- iptables -N SCHREIBAUF
  iptables -A SCHREIBAUF -m limit --limit 4/s -j LOG \
  --log-level info --log-prefix "packet droped "
  iptables -A SCHREIBAUF -j DROP
- iptables -A INPUT -p tcp --dport 515 -j SCHREIBAUF
  - Mit der letzten Zeile wird alles was vorher nur gedropt wurde an die Regel SCHREIBAUF gegeben

Annehmen von Paketen

- iptables -A FORWARD -p tcp -s 192.168.0.0/24 -d 192.168.11.12/32 -j ACCEPT
- -A hänge Regel an die Forward Chain
- -s Source (hier ein Netz)
- Alle tcp Pakete aus dem Netz 192.168.0.0/24 an die IP 192.168.11.12/32 werden erlaubt

Verwendung von Konstanten

- Es ist sicher sinnvoll, am Beginn des Skripts einige Konstanten zu definieren:
  - INTERNET=„eth0“
  - LOOPBACK_INTERFACE=„lo“
  - IPADDR=„<eigene IP>“
  - MY_ISP=„<IP DES SERVICE PROVIDERS>“
  - ...

Aufgabe

- Aufgabe:
- Nur tcp Verbindungen nach aussen, keine eingehenden tcp Verbindungen, ausser auf Port 22 von den Ips 192.168.0.5 und 192.168.0.10

Initialisieren

- Unter Redhat (!)
- /etc/rc5.d/S08iptables
- Link auf entsprechendes Skript
- Im Skript stehen die Filterregeln
- Neue Filterregeln:
- Skript abändern oder neues Skript entsprechend eintragen