



Falko Dressler Tobias Limmer

Netzwerksicherheit WS 2007/08

8. Übungsblatt

15.1.2008



In diesem Übungsblatt wird die Paketfilterimplementierung Netfilter im Linux Kernel am praktischen Beispiel behandelt. Bitte führen Sie die angegebenen Aufgaben der Reihe nach durch!

1. Loggen Sie sich in das Übungsnetz mittels `ssh user@faui7t1.informatik.uni-erlangen.de` und `ssh root@192.168.11.X` auf Ihren Arbeitsrechner ein.

2. Überprüfen Sie die aktuelle Iptables-Konfiguration des Rechners mit dem Kommando

```
iptables [-t <table>] -L
```

Sie interessiert die Standardtabelle “filter” (kein Parameter “-t” nötig) und die Tabelle “nat”.

3. Löschen Sie eventuell vorhandene Regeln mit Hilfe von

```
iptables [-t <table>] -F [<chain>]
```

4. Es ist sinnvoll, Firewallregeln mit Hilfe eines Skripts zu konfigurieren. Legen Sie dazu eine beliebige Datei an und löschen Sie mit den ersten Kommandos alle Ketten (Chains) in den Tabellen “filter” und “nat”. Im Folgenden wird erwartet, dass dieses Skript erweitert wird.

5. Eine kurze Übersicht über den Befehl iptables:

```
iptables <command> <chain> <filter rules> -j <action>
```

- Kommandos: append rule (“-A”), insert rule (“-I”), delete rule (“-D”), replace rule (“-R”), list rules (“-L”), flush rules (“-F”)
- Ketten: INPUT, OUTPUT, FORWARD
- Filterregeln: Ziel-/QuellIP (“-s/-d <ip/mask>”), Empfangs-/Ausgangsschnittstelle (“-i/-o eth1”), Protokoll (“-p <tcp|udp|icmp>”), Quell-/Zielport (“-sport/-dport <port>”, Achtung: nur bei den Protokollen TCP und UDP möglich!)
- Aktionen: ACCEPT, DROP, RETURN, REJECT, Angabe beliebiger benutzerdefinierter Ketten
- Kommandos für Ketten:

- Erzeugen einer Kette: `-N <chain>`
- Löschen einer Kette: `-X <chain>`
- Standardpolicy setzen: `-P <chain> <policy>`
- Beachte: Filterregeln können (fast) beliebig kombiniert werden!

Und dazu noch zwei Beispiele für Connection Tracking:

```
iptables -A INPUT -m state --state RELATED,ESTABLISHED -j ACCEPT
iptables -A INPUT -p tcp --dport 21 -m state --state NEW -j ACCEPT
```

(“-m” spezifiziert die Verwendung eines optionalen Moduls, “state” ist das Connection Tracking-Modul, “-state” gibt den erwarteten Status für eine Verbindung an)

6. Sichern Sie Ihren Rechner ab, sodass auf der Netzwerkschnittstelle “eth1” keine Pakete mehr angenommen werden. Dazu verwerfen Sie alle Pakete in der Kette für einkommende Pakete. ACHTUNG: Sie wollen sich nicht selbst aussperren, deshalb beschränken Sie alle Filterregeln auf Pakete von/zur Schnittstelle “eth1”.
7. Zum Debuggen von Firewallregeln eignet sich der in Netfilter integrierte Zähler. Rufen Sie

```
iptables -L -v
```

dazu auf. Die Zähler können mit

```
iptables -Z
```

zurückgesetzt werden.

8. Versuchen Sie nun, von einem anderen Rechner im Subnetz 192.168.11.0/24 auf ihren Rechner zuzugreifen (z.B. auf Port 80). Beobachten Sie während des Zugriffs die zu den Regeln gehörenden Zähler.
9. Welchen bedeutenden Nachteil besitzt das Ziel DROP? Wie kann dieser umgangen werden? Testen Sie dazu das Ziel “REJECT”.
10. Lassen Sie nun Zugriffe auf Ihren Webserver auf Port 80 zu, alle anderen Ports bleiben gesperrt. Verwenden Sie Connection Tracking.
11. Alle eingetragenen Regeln befinden sich bis jetzt in Kette “INPUT”. Könnte man dies auch in der Kette “OUTPUT” realisieren? Hätte dies Vor- oder Nachteile? Rufen Sie Ihren Betreuer.
12. Lassen Sie Zugriffe mit dem Protokoll ICMP zu. Sie wollen aber keine Pingfloods zulassen, also sollen maximal 10 Pakete in der Sekunde beantwortet werden. Folgender Befehl akzeptiert 10 ICMP-Pakete in der Minute:

```
iptables -A INPUT -p icmp -m limit --limit 10/minute -j ACCEPT
```

Man kann diese Einstellung mit Hilfe der Option “-f” von ping testen.

13. Limitieren Sie die erlaubten Zugriffe auf den Webserver auf 3 pro Minute. Welchen Vorteil bietet die Verwendung von Connection Tracking hierfür? Könnte man dies auch ohne Connection Tracking realisieren? Rufen Sie Ihren Betreuer.
14. Leiten Sie alle einkommenden TCP Pakete an Port 8080 auf Port 80 um.

```
iptables -t nat -A PREROUTING -p tcp --dport 8080 -j DNAT
--to-destination <eigene IP>:80
```

15. Erweitern Sie den verwendeten DNAT-Befehl, sodass alle Pakete an einen anderen Rechner weitergeleitet werden. Achtung: Das Weiterleiten von IP-Paketen muss dazu im Kernel aktiviert sein. Folgender Befehl ermöglicht dies:

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

Testen Sie die Funktionsweise mit Hilfe von tcpdump und rufen Sie den Betreuer.

Diese Woche werden keine Hausaufgaben vergeben!