Hochschule RheinMain, Prof. Dr. Martin Gergeleit

Rechnernetze & Telekommunikation
SoSe 2021
LV 2142

Übungsblatt 8

Bearbeiten Sie diese Aufgaben bitte **vor** Beginn Ihrer Praktikumsgruppe und halten Sie Ihre Ergebnisse **schriftlich** in einem Protokoll Ihrer Versuche fest. Die nötigen Informationen über ACLs erhalten Sie aus den Vorlesungen (https://video.cs.hs-rm.de/course/5/lecture/78/, Rechnernetze und Telekommunikation > 10. Netzwerksicherheit Teil 3), und natürlich im Internet.

Zu Beginn werden Einzelne vom Praktikumsleiter stichprobenartig gebeten elektronisch abzugeben. Die Bearbeitung der Fragen bildet mit eine Grundlage der Bewertung.

Die Fragen werden anschließend in der Praktikumsgruppe interaktiv besprochen und vorgeführt.

Vorbemerkungen und Hinweise

Packet Filtering ist ein Verfahren, das anhand der Informationen im Header eines Paketes entscheidet, ob die Information weiter geleitet wird oder nicht. Die im Header verfügbaren Informationen wie Herkunft, Ziel oder Port (Service) sowie einige andere Informationen werden verwendet, um Regeln zu erstellen für die Weiterleitung oder Abweisung des Paketes. Fast jeder Router bietet heute die Möglichkeit, Packet Filter zu installieren. Das Verfahren ist geeignet, den Datenfluss innerhalb von Netzwerken auf sehr einfache Art zu steuern. Sind die Regeln, die das Packet Filtering beeinflussen, unabhängig vom jeweiligen Netzwerkverkehr und dem Zustand der einzelnen Verbindungen, spricht man von "statischen" Packet Filtern. Der Overhead dieser statischen Packet Filter ist äußerst gering, deshalb liegt die Übertragungsgeschwindigkeit nahe an der Geschwindigkeit der Hardware.

Die Vorteile dieses Verfahrens sind:

- hoher Durchsatz
- billig oder sogar kostenlos

Die Nachteile sind:

- in komplexen Umgebungen wird das Packet Filtering schnell unverwaltbar
- es gibt keine Authentifizierung von Nutzern oder Maschinen

Statische Packet Filter werden aber immer noch genutzt als erste von verschiedenen Sicherheitsmaßnahmen und kombiniert mit dynamischen Packet Filtern, Circuit Gateways oder Application Gateways (Proxies), s. auch Vorlesung.

ACLs auf Cisco Routern

Statische Packet Filter werden auf Cisco Routern durch ACLs (Access Control Lists) implementiert. ACLs können für jedes Interface verschieden sein. Die erweiterten IP ACLs (und nur um diese geht es hier) haben die Nummern 100 bis 199 und werden im globalen Konfigurationsmodus mit:

```
access-list 100 ...
```

konfiguriert. ACLs können auch mit symbolischen Namen benannt werden. Pro Interface kann man je eine ACL für eingehende und ausgehende Pakete konfigurieren. Dies geschieht mit:

```
interface FastEthernet0/1
  ip access-group 100 in ! Filter für eingehende
  ip access-group 101 out ! und ausgehende Pakete
```

Eine ACL enthält null oder mehr `permit` und `deny` Angaben mit einer Bedingung, die der Reihe nach abgearbeitet werden. Die erste zutreffende Bedingung legt die Wirkung der ACL auf ein Paket fest. Trifft eine deny Bedingung zu oder wird das Ende der ACL erreicht, wird das Paket verworfen.

Neue Bedingungen für ACLs werden immer am Ende der Liste eingefügt. Es ist daher fast immer nötig, bei Änderungen die Liste zunächst mit

```
no access-list 100
```

zu entfernen und neu zu schreiben.

Eine leere oder nichtkonfigurierte ACL hat (bei Interface Access Groups) den gleichen Effekt, wie ein generelles deny auf alle Pakete. Aus diesem Grund muss vor Änderungen an ACLs zunächst mit

```
interface FastEthernet 0/1
  no ip access-group 100 in
```

die Konfiguration auf den entsprechenden Interfaces entfernt werden. (Und natürlich nach erfolgter Änderung wieder eingetragen.)

Die Anzahl der Pakete, die auf eine Bedingung zugetroffen haben, kann man sich mit

```
show access-lists 100
```

ansehen.

Um einen ACL-Eintrag Schritt für Schritt aufzubauen geht man wie folgt vor. Das Ziel sei zunächst, email des bekannten Spammers 42.6.6.6 an Rechner in unserem Class C Netz 192.124.255.0/24 zu unterbinden. Wird die ACL Nummer 100 gewählt, muss es also heißen

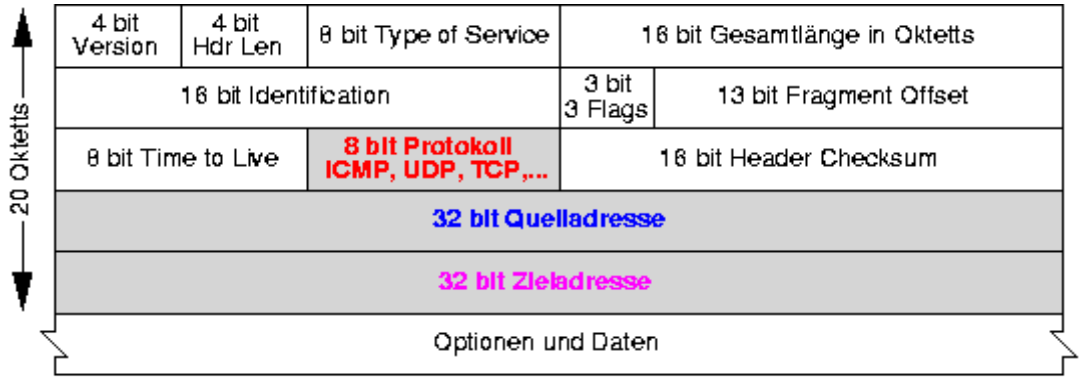
```
access-list 100 ...
```

Danach folgt die Wirkung der noch zu bestimmenden Bedingung, `permit` oder `deny`.

Weil wir an bestimmten Paketen nicht interessiert sind, heißt es also

```
access-list 100 deny ...
```

ACL können entweder auf jedes Paket angewandt werden, das einen Router betritt oder verlässt oder nur auf Pakete eines bestimmten Protokolls. Die Angabe des Protokolls bezieht sich auf das protocol Feld im IP Header (Abbildung 1). Hier muss also eine Zahl von 0 bis 255 angegeben werden; einige wichtige Protokolle kann man auch als Namen angeben: icmp, udp, tcp oder andere. Eine Aufzählung aller Protokolle mit Nummer und Namen findet sich bei der Internet Assigned Number Authority (IANA).



Felder im IP Header

Wenn alle Pakete ungeachtet des Protokolls gefiltert werden sollen, dann wird der Pseudoprotokollnamen ip angegeben.

Da SMTP über TCP implementiert ist, wird als Protokoll `tcp` gewählt:

```
access-list 100 deny tcp ...
```

Als nächstes folgt eine Angabe, die mit der 32bit Quelladresse im IP Header verglichen wird. Dies kann auf drei Arten geschehen:

```
any
```

trifft auf alle Adressen zu

```
host A.B.C.D
```

trifft für genau diese Hostadresse zu

```
A.B.C.D E.F.G.H
```

Trifft zu, wenn die Quelladresse und-verknüpft mit dem Einer-Komplement von E.F.G.H die Adresse A.B.C.D ergibt. Man benutzt diese Form, um Adressbereiche (Netze, Subnetze) zu spezifizieren. Also:

```
access-list 100 deny tcp host 42.6.6.6
```

Äquivalent dazu wäre die Schreibweise

```
access-list 100 deny tcp 42.6.6.6 0.0.0.0
```

Da als Protocol TCP angegeben ist, könnte noch eine Quellport-Nummer folgen. Da der Quellport aber bei SMTP nicht festgelegt ist, entfällt diese Angabe. Ohne diese Angabe trifft die Bedingung auf alle Quellports zu.

Auf die gleiche Art spezifizieren wir die Zieladressen, nämlich unser Class C Netz:

```
access-list 100 deny tcp host 42.6.6.6 192.124.255.0 0.0.0.255
```

Ports werden durch einen Operator und eine Nummer oder Namen spezifiziert. Die Operatoren sind:

lt (less than)
gt (greater than)
eq (equal)
neq (not equal)
range (inclusive range, 2 Nummern folgen)

SMTP ist Port 25, also entweder

```
access-list 100 deny tcp host 42.6.6.6 192.124.255.0 0.0.0.255
eq smtp
```

oder

```
access-list 100 deny tcp host 42.6.6.6 192.124.255.0 0.0.0.255
eq 25
```

Am Ende der Access-Liste steht normalerweise ein implizites

```
access-list 100 deny ip any any
```

was zur Folge hätte, dass wir die gesamte Konnektivität verlören. Wenn wir also lediglich diesen einen Spammer fernhalten wollen, dann sieht die vollständige Access-Liste so aus:

```
access-list 100 deny tcp host 42.6.6.6 192.124.255.0 0.0.0.255
eq smtp
access-list 100 permit ip any any
```

Die Access-Liste müsste dann auf dem Interface zum Internet in eingehender Richtung aktiviert werden.

Rekapitulieren wir: Eine Access-Liste hat die Form

access-list Nr permit|deny Protokoll Quelladresse Zieladresse

Wenn das Protokoll tcp oder udp ist, können optional Portnummern angegeben werden:

access-list Nr permit|deny Protokoll Quelladresse [Quellport] Zieladresse [Zielport]

Wenn das Protokoll tcp ist, kann optional noch 'established' angegeben werden:

access-list Nr permit|deny Protokoll Quelladresse [Quellport] Zieladresse][Zielport] established

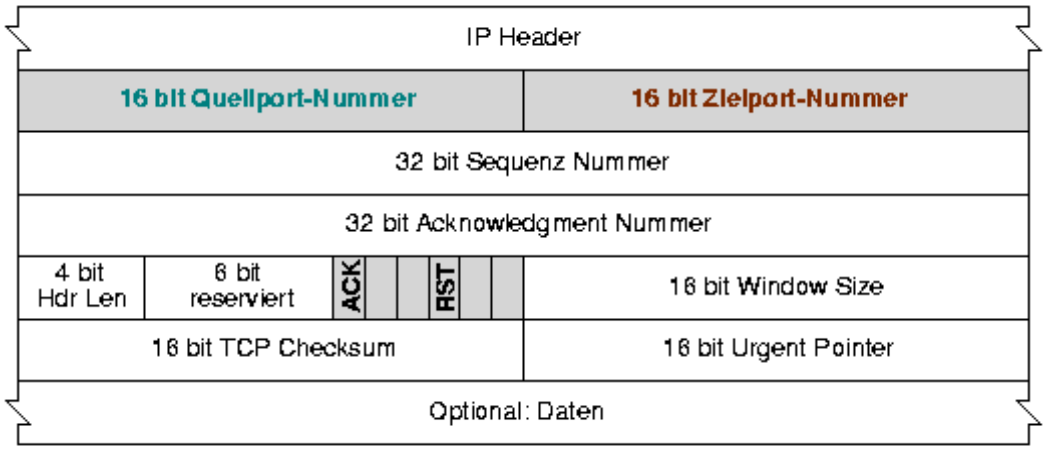
Ein Paket gehört zu einer 'established' TCP Verbindung, wenn das ACK oder RST Bit (oder beide) gesetzt sind. Das erste Paket einer TCP Verbindung hat nur das SYN Bit gesetzt, trifft also nicht auf 'established' zu. Wenn also als erstes eine allgemeine 'established' Bedingung steht:

```
access-list 100 permit tcp any any established
```

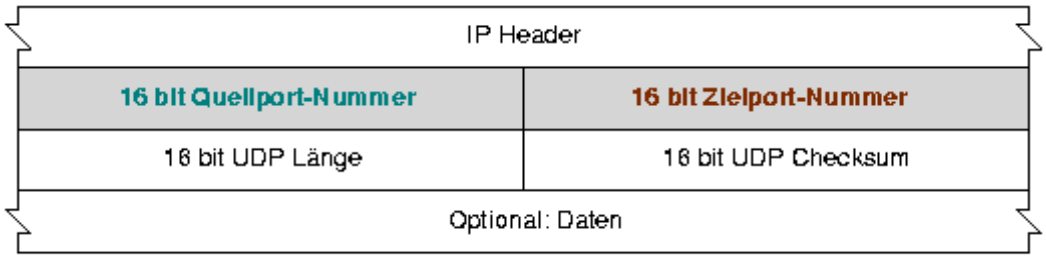
dann müssen alle Pakete etablierter TCP Verbindungen nur mit einem Access-Listen Eintrag verglichen werden. (Dies ist für die Performance bedeutsam, besonders wenn Access-Listen länger werden. Als Faustregel gilt: oft zutreffende Bedingungen möglichst nahe an den Anfang setzen.) Die deny Bedingungen können danach folgen. Wenn wir zum Beispiel eingehendes `telnet` verbieten wollen,

```
access-list 100 permit tcp any any established
access-list 100 deny tcp any 192.124.255.0 0.0.0.255 eq telnet
access-list 100 permit ip any any
```

dann wird das erste (SYN) Paket eines telnet-Versuchs durch den 2. Eintrag verworfen. Eine `telnet`-Verbindung kann somit nie etabliert werden.



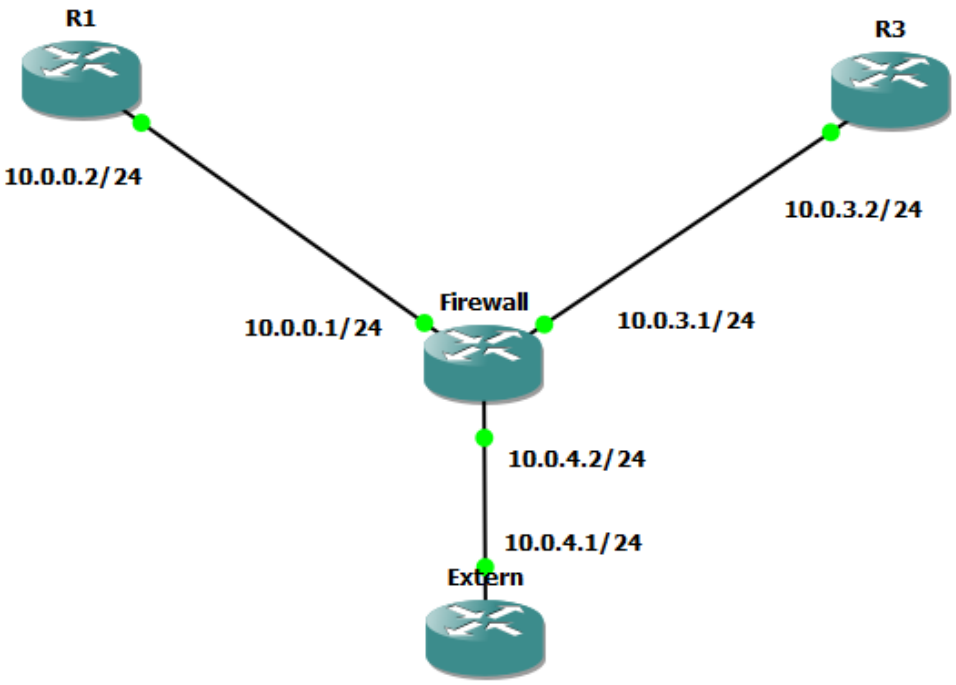
Felder im TCP Header



Felder im UDP Header

Aufgabe 8.1:

Laden Sie die GNS3-Topologie ACL_Setup.gns3project. Sie sollten dann diesen Aufbau sehen:



Testen Sie, dass sich alle Router in der gegebenen Default-Konfiguration an-`ping`-en können.

Konfigurieren Sie auf dem „Firewall“-Router am Interface mit der IP 10.0.4.2 ACLs so, dass der Zugriffe vom Rechner „Extern“ aus auf alle Netze außer dem 10.0.0.0/24 verboten ist, also insbesondere auch auf den Router R3. Testen Sie dies mit `ping`.

Aufgabe 8.2:

Löschen Sie alle ACLs aus 8.1 wieder und testen Sie, ob Sie sich mit dem Password „cisco“ vom Rechner „Extern“ auf allen anderen Routern mit `telnet` einloggen können.

Konfigurieren Sie nun die ACLs so, dass ausschließlich `telnet`-Verbindungen (Port 23 beim Server) zum und vom Router „Extern“ mit allen Routern zugelassen sind (aber z.B. kein `ping` mehr).

Aufgabe 8.3:

Löschen Sie alle ACLs aus 8.2 und konfigurieren und überprüfen Sie nun ACLs, die verhindern, dass TCP-Verbindungen (und damit auch telnet-Sessions) vom Router „Extern“ aus initiiert werden können, wohl aber erlauben, dass die Router R1 und R3 beliebige Clients beliebige TCP-Verbindungen zum „Extern“ aufnehmen können. Unterbinden Sie jeglichen anderen Verkehr und auch Ping-Nachrichten.