14.9 Aufgabe: iptables

Geben sei das iptables-Regelwerk in der Tabelle in Kapitel 10.5.

a) Fertigen Sie ein Netzwerkdiagramm an, das die in diesem Regelwerk erwähnten Hosts und Netze, die notwendigen Datenflüssen inklusive der verwendeten Ports und den Richtungen der Verbindungsinitiierung beinhaltet.

b) Welche der folgenden Zugriffe sind zulässig? Welche davon werden geloggt?

- Ping über ICMP von 192.168.56.1 auf 192.168.56.2
- Ping über ICMP von 192.168.56.10 auf 192.168.56.1
- SSH-Zugriff von 192.168.56.1 auf 192.168.56.2
- SSH-Zugriff von 192.168.56.2 auf 192.168.56.1
- HTTP-Zugriff von 192.168.56.10 auf 192.168.56.2

c) Auf 192.168.56.2 wird ein DNS-Server eingerichtet. Ergänzen Sie passende iptables-Regeln, die den Zugriff von Clients auf diesen Server über eth1 von allen IP-Adressen zulassen.

d) Welche Verbesserungsvorschläge haben Sie bzgl. der vorliegenden Policy?

Port 80 nicht.

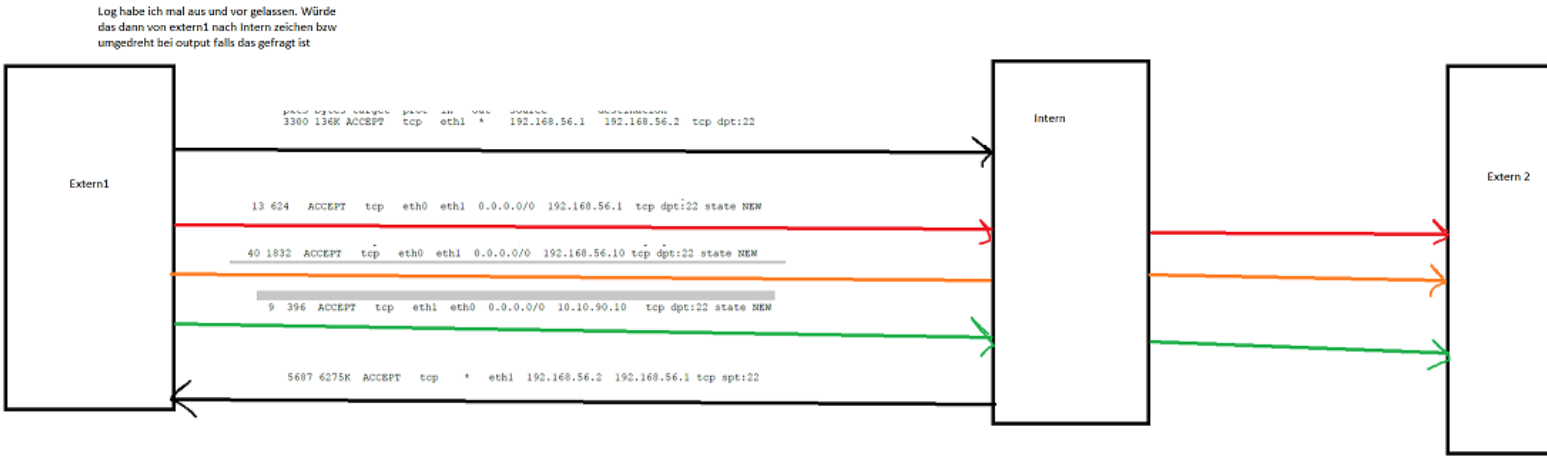
```
Chain INPUT (policy DROP 280 packets, 32685 bytes)
pkts bytes target  prot  in   out   source         destination
3300 136K ACCEPT   tcp   eth1  *     192.168.56.1   192.168.56.2  tcp dpt:22
140 51297 LOG      all   eth0  *     0.0.0.0/0      0.0.0.0/0     LOG flags 0 level4
378K  46M LOG      all   eth1  *     0.0.0.0/0      0.0.0.0/0     LOG flags 0 level4
140 10220 ACCEPT   all   lo    *     0.0.0.0/0      0.0.0.0/0
304 35676 LOG      all   *     *     0.0.0.0/0      0.0.0.0/0     LOG flags 0 level4

Chain FORWARD (policy DROP 0 packets, 0 bytes)
pkts bytes target   prot  in    out   source     destination
4435 1275K LOG      all   eth1  eth0  0.0.0.0/0  0.0.0.0/0     LOG flags 0 level 4
4717 882K  LOG      all   eth0  eth1  0.0.0.0/0  0.0.0.0/0     LOG flags 0 level 4
  13 624   ACCEPT   tcp   eth0  eth1  0.0.0.0/0  192.168.56.1  tcp dpt:22 state NEW
4379 1214K ACCEPT   all   eth1  eth0  0.0.0.0/0  0.0.0.0/0     state REL.,ESTAB.
4609 877K  ACCEPT   all   eth0  eth1  0.0.0.0/0  0.0.0.0/0     state REL.,ESTAB.
   9  396  ACCEPT   tcp   eth1  eth0  0.0.0.0/0  10.10.90.10   tcp dpt:22 state NEW
  40 1832  ACCEPT   tcp   eth0  eth1  0.0.0.0/0  192.168.56.10 tcp dpt:22 state NEW

Chain OUTPUT (policy DROP 7 packets, 588 bytes)
pkts bytes  target   prot   in  out   source         destination
5687 6275K  ACCEPT   tcp    *   eth1  192.168.56.2   192.168.56.1 tcp spt:22
 102 48836  LOG      all    *   eth4  0.0.0.0/0      0.0.0.0/0    LOG flags 0 level4
78904 8127K LOG      all    *   eth1  0.0.0.0/0      0.0.0.0/0    LOG flags 0 level4
 140 10220  ACCEPT   all    *   lo    0.0.0.0/0      0.0.0.0/0
```

Tabelle: Regelwerk und Status von iptables, Ergebnis von `iptables -L -v -n`

a)



PS: Bei Forward steht sehr oft anywhere. Bin mir nicht sicher ob ich was falsch verstehe. Würde sagen das dann super viele Pfeile von Extern 1 nach Intern nach Extern 2 zeigen müssten. Ich glaube ich verstehe aber einfach nur was falsch. Ich spare mir an der Stelle aber die Pfeile. Das Prinzip müsste klar sein. Habe die Konkreteren aufgeschrieben, weil ich bei denen sicherer bin.

b) Welche der folgenden Zugriffe sind zulässig? Welche davon werden geloggt?

- Ping über ICMP von 192.168.56.1 auf 192.168.56.2
- Ping über ICMP von 192.168.56.10 auf 192.168.56.1
- SSH-Zugriff von 192.168.56.1 auf 192.168.56.2
- SSH-Zugriff von 192.168.56.2 auf 192.168.56.1
- HTTP-Zugriff von 192.168.56.10 auf 192.168.56.2

c) Auf 192.168.56.2 wird ein DNS-Server eingerichtet. Ergänzen Sie passende iptables-Regeln, die den Zugriff von Clients auf diesen Server über eth1 von allen IP-Adressen zulassen.

iptables -A INPUT -i eth1 -s anywhere -d 192.168.56.2 -j ACCEPT

d) Wie bereits gesagt habe ich das Gefühl das sehr viele Forwards bestehen die mit anywhere sind. Wenn ich es richtig verstehe, und das extra so ist, dann würde ich das konkretisieren. Mind bei der Destination. Müsste ja sehr viel Aufwand sein. Anywhere Anywhere klingt sehr unpraktisch.