Technische Universität München
Informatik VIII
Netzarchitekturen und Netzdienste
Prof. Dr.-Ing. Georg Carle

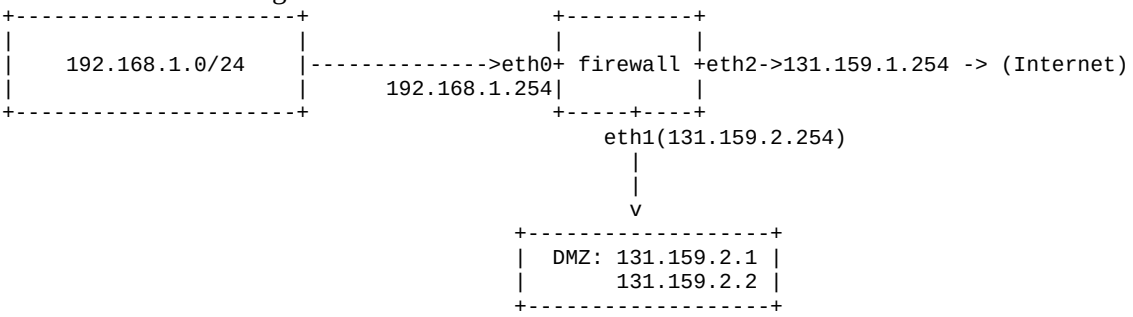
Dipl.-Inform. Heiko Niedermayer
Dipl.-Inform. Ralph Holz
Dipl.-Inform. Lothar Braun

Übung zur Vorlesung „Netzsicherheit“
Übungsblatt 6, WS09/10

Ausgabe:	Mi 20. Jan. 2010
Abgabe:	Do 4. Feb. 2010
Besprechungstermin:	Do 11. Feb. 2010

Aufgabe 1: Firewalls

Sie müssen ein kleines Firmennetzwerk administrieren. Ihr Vorgänger kannte sich nicht so gut mit Paketfiltern aus und hat deswegen einen etwas merkwürdigen Aufbau der Firewall gewählt. Unten sehen sie die Aufteilung der Subnetze:



Bei der Firewall handelt es sich um einen PC mit drei Netzwerkkarten und einer Linux-Installation. Dessen netfilter-Packetfilter wurde wie folgt konfiguriert:

```
iptables -P OUTPUT DROP
iptables -P INPUT DROP
iptables -P FOWARD DROP

iptables -A FORWARD -i eth0 -s 192.168.1.0/24 -j ACCEPT
iptables -t nat -A POSTROUTING -i eth0 -s 192.168.1.0/24 -j MASQUERADE

iptables -A FORWARD -o eth2 -d 131.159.1.1 -p tcp --dport 80 -j ACCEPT
iptables -A FORWARD -o eth2 -d 131.159.1.2 -p tcp --dport 25 -j ACCEPT
```

a) Darf ein Rechner aus 192.168.1.0/24 Verbindungen ins Internet aufbauen? Darf er auf die Dienste in der DMZ zugreifen?

b) Die IP-Adressen von eth1 und eth2 sind statisch, d.h. es liegt keine Dial-up-Verbindung vor, wo das Interface immer wieder bei der Einwahl andere IP-Adressen zugewiesen bekommt. Welche andere Methode NATs zu bauen als per MASQUERADE bietet sich dann an? Warum ist die andere dann vorteilhaft?

c) Angenommen, Sie bauen eine Verbindung ins Internet auf. Kann Ihnen der angesprochene Rechner durch die Firewall antworten?

d) Können Sie die Firewall per SSH administrieren? Funktioniert dies aus dem Internet? Der DMZ? Dem privaten Subnetz?

Aufgabe 2: Intrusion Detection

Das signaturbasierte Intrusion-Detection-System "Snort" (www.snort.org) arbeitet als Netzwerksniffer und untersucht sämtlichen Verkehr, der an einer Netwerkkarte beobachtet werden kann.

a) An welcher Stelle im Netzwerk sollte Snort betrieben werden?

b) Was bedeutet bedeutet es, wenn eine Netzwerkarte im "promiscuous mode" arbeitet? Warum ist dieser Modus fuer Snort wichtig?

Die Konfigurationsdateien von Snort konfigurieren zum einen die verschiedenen Subsysteme von Snort. Zum anderen definieren sie die Regeln, nach denen Snort den Verkehr untersucht.

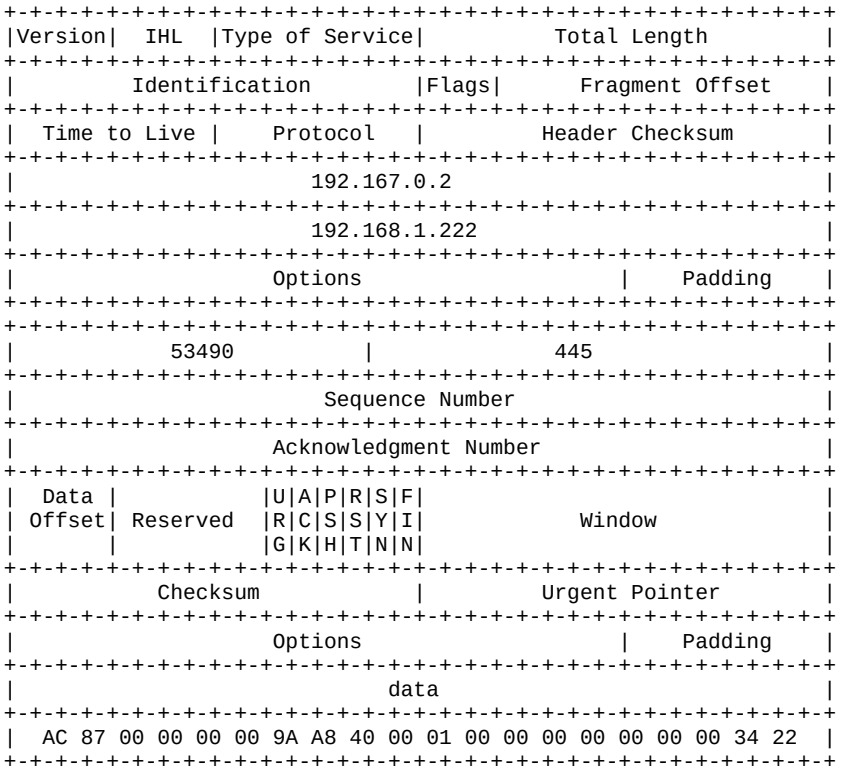
c) Erklären Sie die einzelnen Felder der Regel.

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 445 \
(msg:"ET EXPLOIT MS04011 Lsasrv.dll RPC exploit (Win2k)"; \
content:"|00 00 00 00 9A A8 40 00 01 00 00 00 00 00 00 00|"; classtype: misc-activity; \
reference:url,doc.emergingthreats.net/bin/view/Main/2000046; sid: 2000046; rev:8;)
```

d) Nun komme eine zweite sehr ähnliche Regel in die Regeldatenbank hinzu. Diese sieht wie folgt aus. Warum wird Snort diese Regel ignorieren?

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 445 \
(msg:"ET EXPLOIT MS04011 Lsasrv.dll RPC exploit (WinXP)"; \
content:"|95 14 40 00 03 00 00 00 7C 70 40 00 01|"; classtype: misc-activity; \
reference:url,doc.emergingthreats.net/bin/view/Main/2000033; sid: 2000046; rev:8;)
```

Nun bestehe die Regeldatenbank nur aus der ersten Regel. Die Variable $HOME_NET sei auf 192.168.1.0/24 und die Variable $EXTERNAL_NET auf !$HOME_NET gesetzt. Snort beobachtet nun folgendes Netzwerkpaket (Layer 2 Header wurde der uebersichtlichkeit halber weggelassen). Die nicht genauer spezifizierten Felder seien korrekt gesetzt:



e) Schlägt Snort auf diesem Paket an? Was kann ein Angreifer tun, um einer Erkennung zu entgehen? Gehen Sie davon aus, dass Stream5 und aehnlich arbeitende Snort-Preprozessoren abgeschaltet sind.

--- bitte wenden ---

Aufgabe 3: Angriffe

Beantworten Sie die folgenden Fragen und begründen Sie die Antwort.

a) Sind Flash Crowds ein Denial-of-Service-Angriff?
b) Was wird beim TCP SYN-Flood-Angriff (http://tools.ietf.org/html/rfc4987) angegriffen?
c) Was ist ein Portscan und woran könnten Sie ihn erkennen?
d) Was könnten Sie als Angreifer tun, um trotzdem beim Portscan nicht erkannt zu werden?

Aufgabe 4: Programmierung mit SSL

Nehmen Sie eine Programmiersprache Ihrer Wahl, z.B. Java, Python, ... In dieser Aufgabe sollen Sie eine SSL-Verbindung aufbauen und nutzen. In vielen Fällen finden Sie vorhandenen Code im Internet, den Sie als Basis verwenden können. Dokumentieren Sie die praktischen Teile durch geeignete Code-Fragmente und kurze Erfahrungsberichte.

a) Geben Sie einen kurzen Überblick über die Klassen und Methoden, die ihnen die Sprache oder verwendete Bibliothek gibt, um eine SSL-Verbindung zu verwenden.
b) Der erste praktische Schritt ist die Erzeugung und Speicherung von Schlüssel und Zertifikaten. Hinweise: Dies kann z.B. über ein Keystore geschehen (bei Java über das Werkzeug „keytool“). Achtung, während der Server sein Keystore mit privatem Schlüssel braucht, muss der Client immerhin ein Truststore haben, wo der Schlüssel des Servers bzw. die CA für den Server drin stehen.
c) Bauen Sie nun eine Verbindung zwischen Client und Server auf und lassen Sie in beide Richtungen Texte schicken (z.B. Konsoleneingaben). Tipp: Da viele Socketaufrufe und Streamaufrufe blockierend sind, brauchen Sie ggf. mehrere Threads.
d) Verwenden Sie Wireshark, um Ihre SSL-Verbindung zu überprüfen. Erkennt das Programm das SSL, sind die Daten lesbar?