

Exam fall 2010, INF5004NSA - Intrusion Detection and Firewalls

Read the questions thoroughly before answering and be careful to answer all questions. All written or printed material are allowed. If you think the assignment text is unclear, feel free to state your assumptions in the beginning of your solution.

Problem 1 - Firewalls (30%)

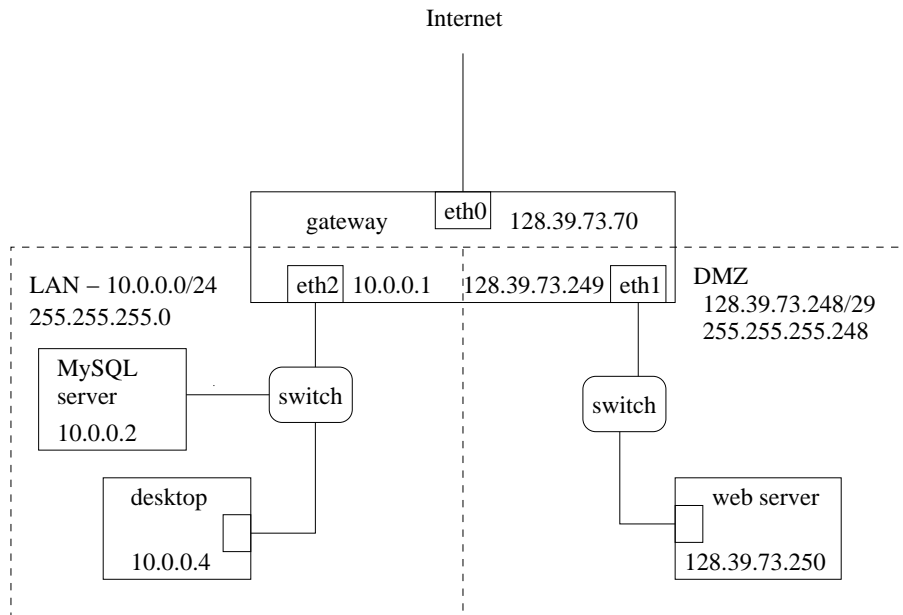


Figure 1: Small network

All the hosts of this small network runs Ubuntu Linux. Assume you have started writing a firewall script at the gateway using Iptables and the start of your `fw.rc` script looks like this:

```
#!/bin/bash
iptables -F
iptables -F -t nat

iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT

iptables --policy INPUT DROP
iptables --policy OUTPUT DROP
iptables --policy FORWARD DROP
```

- What is the purpose of line 5 and 6 and what is the `lo`-interface they refer to? Explain briefly.
- Explain briefly the meaning of a default policy and what it is for this firewall.
- Explain the difference between a stateful and a stateless firewall.

In the following you should make a stateful firewall. Traffic that is not specified should be dropped.

- d) Write rules which makes it possible to connect to the gateway from the desktop using ssh. This should be the only way to connect to the gateway at all.
- e) Add rules so that you are forwarded to the desktop if you connect to the gateway using ssh from the Internet.
- f) Add rules making it possible to surf the web on any public Internet address from any of the hosts in the LAN. Avoid that the public addresses of the DMZ are masqueraded by specifying the source NAT-addresses.
- g) Add rules that makes the web server at 128.39.73.250 available from the Internet and the LAN.
- h) What will be the source IP-address when a packet from the desktop reaches the web server? Explain briefly.
- i) Add rules making it possible to use MSN clients from the desktop.
- j) The DMZ of this small network has public IP's. Discuss the pro and cons of this compared to having private IP's in the DMZ.
- k) Discuss the pro and cons of this way of defining a LAN and DMZ compared to the way they were defined in the home exam (where the DMZ was located between the gateway and the LAN and a choke firewall connected the DMZ and the LAN).

Problem 2 - ssh-scan (20%)

After setting up your firewall you discover that one of the user accounts of the desktop has been compromised by a ssh-scan. This account has not got root privileges.

- a) Explain what a ssh-scan is.
- b) Will the Iptables firewall you have written protect you against a ssh-scan? Explain briefly.
- c) Explain how you can make Iptables protect you against a ssh-scan. Write a rule or rules which protect you if you know how to.
- d) Is it possible to be protected from every single login attempt of a ssh scan? Explain briefly.
- e) How frequent will a public IP typically be attacked?
- f) The attacker now wants to eavesdrop on the traffic to and from the MySQL-server. Will he be able to listen to this traffic using a tool like tcpdump? Explain briefly.
- g) Next the attacker wants to eavesdrop on the traffic between the web server at 128.39.73.250 and the Internet. Will he be able to listen to this traffic using a tool like tcpdump? Explain briefly.
- h) After gaining access to the Linux desktop by the ssh-scan, the hacker sets up a reverse shell using the MSN-port 1863. Explain what a reverse shell is and why a backdoor listening on connections on port 1863 would not help the hacker to reconnect at another time(in case the ssh-password was changed).
- i) Write the Linux commands needed to establish such a reverse shell using netcat.

Problem 3 - Intrusion detection (20%)

After this experience, you would like to write a snort rule which alerts you when someone misuses the MSN-port and sets up a reverse shell. You start investigating and reveals that the only payload

sent by a reverse shell contains the Linux commands and the resulting output. But you realize that it would be inconvenient to write snort rules that match any possible Linux command. Next you discover that the payload of the traffic from the MSN server always starts with three capital letters, like QRY, MSG or ADD. Finally you read the following in the snort manual:

3.5.23 pcre

The pcre keyword allows rules to be written using perl compatible regular expressions.

Format

```
pcre:[!]"(</regex>/|m<delim><regex><delim>)[ismxAEGRUBPHMCOIDKYS]";
```

- a) Based on this knowledge, write a snort rule which raises an alert when a MSN connection is used for non-MSN traffic, like a reverse shell does. Assume you run snort at the gateway. Explain by words what the pcre should do if you do not know the syntax. Explain briefly the most important parts of your rule and your strategy when writing this rule.
- b) Explain how you would test your new rule for false positives and false negatives.
- c) If the attacker knew the contents of your rule preventing against his reverse shell, how could he change his strategy in order to evade your rule?

The following is from the snort manual:

Event Filters

You can use event filters to reduce the number of logged events for noisy rules.

This can be tuned to significantly reduce false alarms.

- d) Explain briefly if and how this could be relevant for your reverse shell detection rule.

Problem 4 - Stimulus and response (10%)

Explain the first packet of response you will get when you try to connect to a server using TCP to port 123 and

- a) the corresponding service is running and there is no firewall.
- b) there is no corresponding service running and there is no firewall.
- c) your connection is stopped by a firewall.

Explain the first packet of response you will get when you try to connect to a server using UDP to port 123 and

- d) the corresponding service is running and there is no firewall.
- e) there is no corresponding service running and there is no firewall.
- f) your connection is stopped by a firewall.

Problem 5 - Spoofing (20%)

At the host rex with IP 128.39.89.9 you do the following:

```
rex# hping3 -c 1 nexus2.iu.hio.no -p 80 -S --spooof fw10.vlab.iu.hio.no
HPING nexus2.iu.hio.no (eth0 128.39.89.23): S set, 40 headers + 0 data bytes
```

```
1 packets transmitted, 0 packets received, 100% packet loss
```

a) Explain what kind of packet is sent, focusing on the IP-addresses. The IP at fw10.vlab.iu.hio.no is 128.39.73.160, but you may refer to it as the fw10-IP or just fw10.

b) Explain the output below from tcpdump at nexus2 during the same event:

```
nexus2# tcpdump port 80
01:39:18.901525 IP fw10.vlab.iu.hio.no.2269 > nexus2.iu.hio.no.www: S 113734219:113734219(0)
01:39:18.901570 IP nexus2.iu.hio.no.www > fw10.vlab.iu.hio.no.2269: S 1901083161:1901083161(0) ack 113734220
01:39:18.901956 IP fw10.vlab.iu.hio.no.2269 > nexus2.iu.hio.no.www: R 113734220:113734220(0)
```

c) Explain what you would expect to see when simultaneously running tcpdump at the spoofed host fw10:

```
fw10# tcpdump port 80
```

d) Explain what you would expect to see when simultaneously running tcpdump at the host rex, the host you ran hping3 from:

```
rex# tcpdump port 80
```

e) Assume there is no udp server running at port 80 on the target nexus2 and you spoof using the following command:

```
rex# hping3 -c 1 nexus2.iu.hio.no --udp -p 80 --spooof fw10.vlab.iu.hio.no
```

Explain what you would expect to see if you run tcpdump at the spoofed address:

```
fw10# tcpdump host nexus2.iu.hio.no
```

f) In what ways can a gateway firewall contribute to reducing the number of spoofed IP-addresses?

g) When a server receives a SYN-packet and replays by a SYN-ACK it will consume some memory and wait for the ACK which establishes the connection. When SYN-flooding a target, the attacker takes up all these resources by flooding the server with SYN messages. It is common to use spoofed addresses when SYN-flooding. However, why will this attack not be effective when using any IP, like the one chosen in the example of question a)? What kind of spoofed IP should be chosen in order to make the attack effective?

h) Explain the output of the following tcpdump and try to figure out what caused this traffic:

```
nix:~# tcpdump -vv icmp
tcpdump: listening on eth0, link-type EN10MB (Ethernet), capture size 96 bytes
00:38:13.514957 IP (ttl 64, id 54839) host1.interbusiness.it > nix.iu.hio.no: icmp 8: echo request seq 0
00:38:13.514988 IP (ttl 64, id 65241) nix.iu.hio.no > host1.interbusiness.it: icmp 8: echo reply seq 0
00:38:14.515610 IP (ttl 64, id 24211) tid.uio.no > nix.iu.hio.no: icmp 8: echo request seq 256
00:38:14.515632 IP (ttl 64, id 56865) nix.iu.hio.no > tid.uio.no: icmp 8: echo reply seq 256
00:38:15.516512 IP (ttl 64, id 61874) softbank.bbtec.net > nix.iu.hio.no: icmp 8: echo request seq 512
00:38:15.516546 IP (ttl 64, id 37892) nix.iu.hio.no > softbank.bbtec.net: icmp 8: echo replyseq 512
```