Global Information Assurance Certification Paper

SANS SNAP Intrusion Detection Assignment (San Jose 2000)

By: Shane Akhgar

Detect 1

May 23 12:57:52 my kernel: Packet log: input REJECT eth0 PROTO=6 64.228.196.171:62393 my.ids.box:136 L=60 S=0x00 I=61728 F=0x4000 T=51 SYN (#7)
May 23 12:58:07 my kernel: Packet log: input REJECT eth0 PROTO=6 64.228.196.171:62394 my.ids.box:82 L=60 S=0x00 I=61729 F=0x4000 T=51 SYN (#7)
May 23 12:58:22 my kernel: Packet log: input REJECT eth0 PROTO=6 64.228.196.171:62395 my.ids.box:1362 L=60 S=0x00 I=61730 F=0x4000 T=51 SYN (#7)
May 23 12:58:37 my kernel: Packet log: input REJECT eth0 PROTO=6 64.228.196.171:62396 my.ids.box:589 L=60 S=0x00 I=61731 F=0x4000 T=51 SYN (#7)
May 23 12:58:52 my kernel: Packet log: input REJECT eth0 PROTO=6 64.228.196.171:62397 my.ids.box:124 L=60 S=0x00 I=61732 F=0x4000 T=51 SYN (#7)
May 23 12:59:07 my kernel: Packet log: input REJECT eth0 PROTO=6 64.228.196.171:62398 my.ids.box:802 L=60 S=0x00 I=61733 F=0x4000 T=51 SYN (#7)
May 23 12:59:22 my kernel: Packet log: input REJECT eth0 PROTO=6 64.228.196.171:62399 my.ids.box:4500 L=60 S=0x00 I=61734 F=0x4000 T=51 SYN (#7)
May 23 12:59:37 my kernel: Packet log: input REJECT eth0 PROTO=6 64.228.196.171:62400 my.ids.box:258 L=60 S=0x00 I=61735 F=0x4000 T=51 SYN (#7)
May 23 12:59:52 my kernel: Packet log: input REJECT eth0 PROTO=6 64.228.196.171:62401 my.ids.box:282 L=60 S=0x00 I=61736 F=0x4000 T=51 SYN (#7)
May 23 13:00:07 my kernel: Packet log: input REJECT eth0 PROTO=6 64.228.196.171:62402 my.ids.box:1444 L=60 S=0x00 I=61737 F=0x4000 T=51 SYN (#7)

1. Source of trace: My network

2. Detect generated by: ipchains log

   Explanation of fields:

   May 23 12:58:52 (date&timestamp) my (hostname) kernel: Packet log: (log source & type) input (rulechain) REJECT (rule action) eth0 (interface) PROTO=6 (protocol) 64.228.196.171:62397 (source ip:source port) my.ids.box:124 (destination ip:dest port) L=60 (packet length) S=0x00 (Type Of Service) I=61732 (datagram ID) F=0x4000 (fragment offset) T=51 (TTL) SYN (flags) (#7) (triggering rule)

3. Probability of spoofed source address: Possible, but unlikely. Appears to be a standard tcp scan. Source address is a local ISP's DSL network.

4. Description of attack: Host scan for open ports. Standard reconnaissance. The attacker seems be making an effort to be subtle, since each connection attempt is 15 seconds apart. This is the default behaviour for the nmap scanner, when called with the "-T Sneaky" option.

5. Attack mechanism: The scanner is attempting to open tcp connections to various ports on the target host.

6. Correlations: Very common scanning method.

7. Evidence of active targeting: Probably a result of a previous pingscan to find active hosts. The target box in this case has no other function on the network, other than as a honeypot/IDS unit.

8. Severity: (critical + lethal) - (system + net countermeasures)

   (5 + 2) - (5 + 4) = -2
   Explanation:
   Critical=5 : It's an important machine.
   Lethal=2 : The scan itself is not particularly lethal, but could potentially lead to other problems.
   System=5 : It's a well-secured, locked down, bastion host.
   Countermeasures=4 : It's very selective about what connections it allows, but there is some, limited, external access (ssh).

9. Defensive recommendation: There's no risk to the system in question, but the fact that the attacker hit this machine means they have been scanning the whole subnet. We may want to contact the ISP with log snippets and have them give the user the traditional smack on the wrist.

10. Multiple choice question:

    How would you describe this trace:
    a) inverse mapping attempt
    b) tcp port scan of a specific host
    c) udp port scan of a specific host
    d) buffer overflow attack
    Answer: (b)

Detect 2:

Mar 27 13:56:14 unit1 kernel: Packet log: statinpt REJECT eth0 PROTO=6 61.137.156.46:2879 my.ip.net.18:8080 L=48 S=0x00 I=13082 F=0x4000 T=109 SYN (#53)
Mar 27 13:56:17 unit1 kernel: Packet log: statinpt REJECT eth0 PROTO=6 61.137.156.46:2879 my.ip.net.18:8080 L=48 S=0x00 I=33818 F=0x4000 T=109 SYN (#53)
Mar 27 13:56:14 unit2 kernel: Packet log: statinpt REJECT eth2 PROTO=6 61.137.156.46:2881 my.ip.net.20:8080 L=48 S=0x00 I=13594 F=0x4000 T=109 SYN (#62)

Mar 27 13:56:17 unit2 kernel: Packet log: statinpt REJECT eth2 PROTO=6 61.137.156.46:2881 my.ip.net.20:8080 L=48 S=0x00 I=34074 F=0x4000 T=109 SYN (#62)
Mar 27 13:56:13 unit3 kernel: Packet log: input REJECT eth0 PROTO=6 61.137.156.46:2870 my.ip.net.9:8080 L=48 S=0x00 I=3610 F=0x4000 T=108 SYN (#7)
Mar 27 13:56:16 unit3 kernel: Packet log: input REJECT eth0 PROTO=6 61.137.156.46:2870 my.ip.net.9:8080 L=48 S=0x00 I=31514 F=0x4000 T=108 SYN (#7)

1. Source of trace: My network

2. Detect generated by: ipchains logs

   This scan was picked up by multiple machines. This indicates that the attacker was scanning the subnet.
   Explanation of fields: Here is an example, with explanations inserted after each field (in parentheses).
   May 23 12:58:52 (date&timestamp) my (hostname) kernel: Packet log: (log source & type) input (rulechain) REJECT (rule action) eth0 (interface) PROTO=6 (protocol) 64.228.196.171:62397 (source ip:source port) my.ids.box:124 (destination ip:dest port) L=60 (packet length) S=0x00 (Type Of Service) I=61732 (datagram ID) F=0x4000 (fragment offset) T=51 (TTL) SYN (flags) (#7) (triggering rule)

3. Probability of spoofed source address: Unlikely. The attacker needs the replies, for this scan to be of any use.

4. Description of attack: Attacker is scouting the whole subnet for either:

   a) web proxies/caches (typically on port 8080), or
   b) the RingZero trojan

5. Attack mechanism:

   a) In the case of web proxies, the attacker might be looking for proxy servers with known exploits. For an example, see CVE-1999-0710, which involves the cachemgr.cgi shipped with some Redhat squid installations.
   b) In the case of the RingZero trojan, the attacker would be looking for Windows hosts that have been infected with this trojan, which they could then remotely take over and control remotely.

6. Correlations: RingZero scans seem to be increasingly common, of late. Squid and other proxy exploits are not quite so frequent, but unpatched or older installations still exist, and attackers still do look for them.

7. Evidence of active targeting: Since the scan was logged by several machines on that subnet, that would indicate the attacker is not targeting a specific host. Also, there are no Windows hosts on that subnet, eliminating the possibility of a targetted RingZero scan. Nor are there any webservers.

8. Severity: (2 + 3) - (5 + 5) = -5

   While this attack has some potential, none of the hosts targetted are vulnerable to it, and all firewalls

are blocking that port.

9. Defensive recommendation: All systems are considered safe from this attack.

10. Multiple choice question: Which of the following can you eliminate as a possible intention for the above scan:

    a) scanning to find Windows machines infected by RingZero
    b) scanning for Solaris systems running tooltalk
    c) scanning for web proxies
    d) scanning for Squid
    Answer: (b)

Detect 3:

*Jun 5 00:01:30 EST: %SEC-6-IPACCESSLOGP: list 104 denied udp 192.168.1.1(0) -> my.web.srvr.193(0), 1 packet
*Jun 5 00:05:27 EST: %SEC-6-IPACCESSLOGP: list 104 denied tcp 192.168.1.226(0) -> my.web.srvr.193(0), 1 packet

1. Source of trace: My network

2. Detect generated by: Cisco ACL logs.

   Explanation of format:
   *Jun 5 00:01:30 EST: (date & timestamp) %SEC-6-IPACCESSLOGP: list 104 (router access list) denied (action) udp (protocol) 192.168.1.1(0) (source address and port) ->my.web.srvr.193(0) (destination address and port), 1 packet

3. Probability of spoofed source address: Certain. This ACL blocks incoming packets from the Internet. The source address is an RFC1918 non-routable IP.

4. Description of attack: These packets are clearly crafted. The source and destination ports are both 0. The traffic is probably also source routed. The attacker is probably trying to slip these packets past a firewall, by making it believe that the source of the traffic is internal. What makes this scary is there is an internal network that uses that range of addresses (but not those particular ones). It's possible that the attacker has somehow found this out.

5. Attack mechanism: Send packets with "trusted" source addresses, in the hopes of sneaking them past security gateways. To ensure that the packets return to the sender, they must be using source routing.

6. Correlations: While source routing is relatively uncommon in today's Internet, it can still be an effective attack mechanism. Particularly when combined with information gathered from other reconnaissance.

7. Evidence of active targeting: Certain. That machine in particular is a well known web server. Only that machine was targeted by this attack.

8. Severity: (3 + 3) - (4 + 5) = -3

   Explanation: The attack can have serious results, potentially. While the targetted host is not necessarily vulnerable (although it is an NT box) the router blocks all packets claiming to be from the internal network at the point of entry.

9. Defensive recommendation: While defenses are adequate to prevent this particular attack, it is still worrisome. It shows an attacker with a level of sophistication above your average script-kiddy, who seems to be armed with some possibly useful information. Increased watchfulness recommended.

10. Multiple choice question: Which of the following is NOT applicable to this trace:

    a) tcp source and destination ports both zero indicate a crafted packet
    b) the source addresses indicate crafted packets
    c) the attacker is attempting to port scan the web server
    d) the packets are not being received by the web server
    Answer: (c)

Detect 4:

[**] Psyber Stream [**]
05/20-01:14:38.032403 64.228.194.66:53 -> my.home.ip:1170
UDP TTL:63 TOS:0x0 ID:65313
Len: 235

1. Source of trace: My home network

2. Detect was generated by: Snort v1.6 running on a gateway machine I have connected to my DSL provider. This machine does NAT for my internal home network.

3. Probability of spoofed source address: Unlikely. The source address is that of another DSL user, from the same provider.

4. Description of attack: Psyber Stream is another in the endless list of remote control Windows trojans. The attacker was likely looking for a machine that was infected.

5. Attack mechanism: This particular trojan allows the attacker to receive the audio input of the victim's microphone! Interesting. The other interesting thing about this is that the source port is port 53, the DNS port. There is a faint possibility that the machine with the source IP *is* in fact a nameserver, and was attempting to return a query.

6. Correlations: Probably not a common trojan, but it certainly appears on all the trojan port lists.

7. Evidence of active targeting: Likely. My gateway had just come up and received this IP. I suspect that I got the IP of someone whose windows machine had been infected. The attacker was probably trying to re-establish contact with their victim, who may have recently rebooted or shut down.

8. Severity: (5 + 3) - (5 + 5) = -2

   It's my firewall/gateway, which is vital (at least to me). But I don't run Windows, and this attack is pointless against that machine.

9. Defensive recommendation: Defenses are fine.

   Multiple choice question: If the source address were in fact spoofed, what would be the point?
   a) to frame, or incriminate the person who actually has the address
   b) to guarantee that the packet would get through a firewall or ids
   c) to fool the host with the trojan on it, to get it to answer
   d) inverse mapping
   Answer: (a)

Detect 5:

Jun 12 15:53:31 nat.fw kernel: Packet log: statinpt REJECT eth0 PROTO=6 207.253.211.198:1413 nat.fw:31337 L=48 S=0x00 I=50927 F=0x4000 T=122 SYN (#53)
Jun 12 15:53:34 nat.fw kernel: Packet log: statinpt REJECT eth0 PROTO=6 207.253.211.198:1413 nat.fw:31337 L=48 S=0x00 I=55279 F=0x4000 T=122 SYN (#53)
Jun 12 15:53:40 nat.fw kernel: Packet log: statinpt REJECT eth0 PROTO=6 207.253.211.198:1413 nat.fw:31337 L=48 S=0x00 I=60911 F=0x4000 T=122 SYN (#53)
Jun 12 15:53:52 nat.fw kernel: Packet log: statinpt REJECT eth0 PROTO=6 207.253.211.198:1413 nat.fw:31337 L=48 S=0x00 I=2032 F=0x4000 T=122 SYN (#53)

1. Souce of trace: My network

2. Detect generated by: linux ipchains log

3. Probability of spoofed source address: Unlikely, but possible. This looks like a scan for Back Orifice. The source IP is a ppp dialup from local ISP.

4. Description of attack: The attacker is looking for Windows hosts that are infected with the Back Orifice remote control trojan.

5. Attack mechanism: Upon receiving a reply from an infected host, the attacker can attempt to take control of the machine, remotely.

6. Correlations: Back Orifice is probably one of the most well known trojans. There are undoubtedly hundreds, if not thousands, of infected hosts out there.

7. Evidence of active targeting:
   Since the reporting machine is a linux machine, the answer would ordinarily be a no. But that machine also happens to be the masquerading gateway for a subnet that contains a number of Windows hosts. It's possible that one of those hosts was infected, and sent out a report to its "master", via a Back Orifice plugin such as "Butt Trumpet". The "master" would see the source IP as that of the gateway, not the infected host, and try to reply to it instead.

8. Severity: (2 + 5) - (3 + 5) = -1

9. Defensive recommendations: The firewall has clearly blocked the returning packets, but the internal Windows boxes need to be checked for evidence of BO. Unfortunately, on the subnet this trace comes from, there was no corresponding internal IDS system to show traces of the _outgoing_ packets, if any.

10. Multiple choice question: What would happen if the gateway or firewall was NOT running NAT, and the Windows machines had publicly routable addresses?
    a) the firewall would crash
    b) the outgoing packets would give away the address of the victim, and the attacker could try to contact it directly
    c) Windows would crash
    d) Back Orifice would crash
    Answer: (b), although given that (c) is an inevitable fact of life, points may be given for that answer too. ;)

Detect 6:

May 29 11:36:53 nat.fw kernel: Packet log: statinpt REJECT eth2 PROTO=6 194.79.168.3:4967 nat.fw:1080 L=60 S=0x00 I=62842 F=0x4000 T=50 SYN (#62)

May 29 11:36:56 nat.fw kernel: Packet log: statinpt REJECT eth2 PROTO=6 194.79.168.3:4967 nat.fw:1080 L=60 S=0x00 I=62881 F=0x4000 T=50 SYN (#62)
May 29 11:37:02 nat.fw kernel: Packet log: statinpt REJECT eth2 PROTO=6 194.79.168.3:4967 nat.fw:1080 L=60 S=0x00 I=62918 F=0x4000 T=50 SYN (#62)
[snip]
May 29 11:56:01 nat.fw kernel: Packet log: statinpt REJECT eth2 PROTO=6 194.79.168.3:4967 nat.fw:1080 L=60 S=0x00 I=7035 F=0x4000 T=49 SYN (#62)
May 29 11:58:01 nat.fw kernel: Packet log: statinpt REJECT eth2 PROTO=6 194.79.168.3:4967 nat.fw:1080 L=60 S=0x00 I=7668 F=0x4000 T=50 SYN (#62)
May 29 12:00:01 nat.fw kernel: Packet log: statinpt REJECT eth2 PROTO=6 194.79.168.3:4967 nat.fw:1080 L=60 S=0x00 I=8146 F=0x4000 T=49 SYN (#62)
May 29 13:31:40 nat.fw kernel: Packet log: statinpt REJECT eth2 PROTO=6 194.79.168.3:2035 nat.fw:1080 L=60 S=0x00 I=42197 F=0x4000 T=50 SYN (#62)
[snip]
May 29 14:02:52 nat.fw kernel: Packet log: statinpt REJECT eth2 PROTO=6 194.79.168.3:2114 nat.fw:1080 L=60 S=0x00 I=55270 F=0x4000 T=49 SYN (#62)

1. Source of trace: My network

2. Detect generated by: linux ipchains log

3. Probability of spoofed source address: Unlikely. The ip is part of a small subnet (/28, ie: block of 16) addresses allocated to a company in France. This information was found through querying the RIPE whois server.

4. Description of attack: The attacker is possibly looking for a SOCKS server, since that is usually what runs on 1080. A quick glance at the Windows trojan list shows that a trojan known as WinHole also lives on that port, however. If it were a simple socks scan, one would think that the attacker would give up after the first few rejects from the firewall/gateway. But these traces show up starting at 11:36, continue till about noon, then start again at 13:30 and continue till 14:02. That doesn't fit the profile of a scan. It's more likely someone attempting to talk to a trojan-infected Windows host on the other side of the NAT gateway.

5. Attack mechanism: WinHole is apparently a trojan that turns an infected Windows box into a gateway. Lovely.

6. Correlations: Uncommon, but seems like a potentially very useful trojan for an attacker.

7. Evidence of active targeting: Given that the attacker was trying for over two hours, the odds are that they were actively targeting a system on my network.
8. Severity: (2 + 3) - (3 + 5) = -3

9. Defensive recommendations: Once again, although the firewall seems to be blocking communication between the attacker and their intended victim, it would be advisable to scan the Windows hosts on that subnet for trojans.

10. Multiple choice question: What is SOCKS used for?
    a) It's a firewall
    b) It's a remote administration tool
    c) It's a packet filter
    d) It's an application proxy
    Answer: (d)

Detect 7:

*Jun 7 15:02:42 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.0.0.2 -> my.ip.net.243 (0/0), 1 packet
*Jun 7 15:05:22 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.0.0.2 -> my.ip.net.18 (0/0), 1 packet
*Jun 7 15:06:39 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.0.242.30 -> my.ip.net.66 (0/0), 1 packet
*Jun 7 15:09:27 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.0.0.2 -> my.ip.net.20 (0/0), 1 packet
*Jun 7 15:11:10 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.0.0.2 -> my.ip.net.242 (0/0), 1 packet
*Jun 7 15:11:12 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.0.0.2 -> my.ip.net.241 (0/0), 1 packet
*Jun 7 15:11:35 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.0.0.2 -> my.ip.net.225 (0/0), 1 packet
*Jun 7 15:11:42 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.0.0.2 -> my.ip.net.228 (0/0), 1 packet
*Jun 7 15:11:45 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.0.0.2 -> my.ip.net.226 (0/0), 1 packet
*Jun 7 15:13:40 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.10.11.2 -> my.ip.net.18 (0/0), 1 packet
*Jun 7 15:13:41 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.10.11.2 -> my.ip.net.243 (0/0), 1 packet
*Jun 7 15:14:02 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.252.32.4 -> my.ip.net.18 (0/0), 1 packet
*Jun 7 15:14:07 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.252.32.4 -> my.ip.net.243 (0/0), 1 packet
*Jun 7 15:14:13 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.252.27.102 -> my.ip.net.243 (0/0), 1 packet
*Jun 7 15:16:07 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.0.0.2 -> my.ip.net.18 (0/0), 1 packet
*Jun 7 15:17:01 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.0.0.2 -> my.ip.net.243 (0/0), 1 packet
*Jun 7 15:18:07 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.252.32.4 -> my.ip.net.20 (0/0),

1 packet
*Jun 7 15:18:13 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.252.27.102 -> my.ip.net.20 (0/0), 1 packet
*Jun 7 15:20:14 EST: %SEC-6-IPACCESSLOGDP: list 104 denied icmp 10.0.0.2 -> my.ip.net.20 (0/0),1 packet

1. Source of trace: My network

2. Detect generated by: Cisco ACL

3. Probability of spoofed source address: Pretty certain. The source addresses are RFC1918 non-routable IPs.

4. Description of attack: Reconnaissance attack using spoofed addresses? Useful if combined with source routing. The range and distribution of both the source and destination addresses is somewhat strange, however.

5. Attack mechanism: Seems to be some sort of mapping attempt. Or perhaps an attempt to see what packets can be slipped through filters.

6. Correlations: Unknown.

7. Evidence of active targeting: Given the timing with which the packets are arriving, and the strange distribution, I'm considering the possibility that its the result of a routing error on the part of my provider. Or perhaps simply their internal monitoring systems? Large ISPs sometimes use RFC1918 addresses for internal routing.

8. Severity: (3 + 1) - (5 + 5) = -6
   Not about to succeed in any way. The border router is simply dropping all packets with those source addresses.

9. Defensive recommendations: Defenses are fine, but might want to check with ISP...

10. Multiple choice test question: Why are 10.x.x.x addresses not publicly routable?
    a) It's too small a number. Routers reduce the ttl and drop the packets.
    b) Firewalls all around the Internet are configured to block those addresses.
    c) It's not a real IP address.
    d) The IETF labelled them "reserved".
    Answer: (d)

Detect 8:

Jun 2 19:12:49 unit1 kernel: Packet log: input REJECT eth0 PROTO=6 24.113.3.55:3675 my.ip.net.9:21 L=48 S=0x00 I=29186 F=0x4000 T=49 SYN (#7)
Jun 2 19:12:52 unit1 kernel: Packet log: input REJECT eth0 PROTO=6 24.113.3.55:3675 my.ip.net.9:21 L=48 S=0x00 I=54530 F=0x4000 T=49 SYN (#7)
Jun 2 19:12:50 unit2 kernel: Packet log: statinpt REJECT eth2 PROTO=6 24.113.3.55:3686 my.ip.net.20:21 L=48 S=0x08 I=38914 F=0x4000 T=50 SYN (#61)
Jun 2 19:12:53 unit2 kernel: Packet log: statinpt REJECT eth2 PROTO=6 24.113.3.55:3686 my.ip.net.20:21 L=48 S=0x08 I=61442 F=0x4000 T=50 SYN (#61)
Jun 2 19:12:50 unit3 kernel: Packet log: statinpt REJECT eth0 PROTO=6 24.113.3.55:3684 my.ip.net.18:21 L=48 S=0x08 I=38146 F=0x4000 T=50 SYN (#53)
Jun 2 19:12:53 unit3 kernel: Packet log: statinpt REJECT eth0 PROTO=6 24.113.3.55:3684 my.ip.net.18:21 L=48 S=0x08 I=60674 F=0x4000 T=50 SYN (#53)

1. Source of trace: My network

2. Detect generated by: linux ipchains logs (multiple hosts): This scan was picked up by multiple machines. This indicates that the attacker was scanning the subnet.

3. Probability of spoofed source addresses: Possible, unlikely. Source address is a cable modem.

4. Description of attack: The attacker is scanning for listening ftp servers. There are none on that network.

5. Attack mechanism: This is a reconnaissance scan, to find ftp servers, which would presumably be followed by an attempt to determine the type of server, followed by attempting various ftp exploits, of which there have been a great many lately.

6. Correlations: In the past few months, there have been a great many ftp exploits announced (on Bugtraq, etc). It's not surprising to see more people looking for servers to exploit.

7. Evidence of active targeting: There are no ftp servers available on that net, so it would appear the attacker is "just fishing".

8. Severity: (3 + 1) - (4 + 4) = -4

9. Defensive recommendation: No danger here.

10. Multiple choice question: What would happen if there were an ftp server, but behind one of the firewalls, which allowed through connections to port 21?
    a) FTP doesn't run on port 21!
    b) The firewall would stop all attempts to exploit the ftp server
    c) The firewall is either configured to allow the packets or it isn't. Most packet filters don't check content.
    d) The ftp server wouldn't be able to reply to the attacker.
    Answer: (c)

Detect 9:

19:42:09 drop dat >qfe0 proto udp src 64.228.196.171 dst an.other.web.server service 472 s_port 62748 len 28 rule 11
19:42:09 drop dat >qfe0 proto udp src 64.228.196.171 dst an.other.web.server service 997 s_port 62748 len 28 rule 11
19:42:09 drop dat >qfe0 proto udp src 64.228.196.171 dst an.other.web.server service 889 s_port 62748 len 28 rule 11
19:42:09 drop dat >qfe0 proto udp src 64.228.196.171 dst an.other.web.server service 439 s_port 62748 len 28 rule 11
19:42:09 drop dat >qfe0 proto udp src 64.228.196.171 dst an.other.web.server service 1438 s_port 62748 len 28 rule 11
19:42:09 drop dat >qfe0 proto udp src 64.228.196.171 dst an.other.web.server service 1456 s_port 62748 len 28 rule 11
19:42:09 drop dat >qfe0 proto udp src 64.228.196.171 dst an.other.web.server service 97 s_port 62748 len 28 rule 11
19:42:09 drop dat >qfe0 proto udp src 64.228.196.171 dst an.other.web.server service 1212 s_port 62749 len 28 rule 11
19:42:09 drop dat >qfe0 proto udp src 64.228.196.171 dst an.other.web.server service 1366 s_port 62749 len 28 rule 11

1. Source of trace: My network

2. Detect generated by: Checkpoint Firewall-1 v4.0

3. Probability of spoofed source addresses: Unlikely. Appears to be a udp port scan. The source IP is the same as the one seen in Detect #1. The attacker seems to have taken a great interest in my network on that date. Although this scan seems to be several hours after the other one. Perhaps they were busy elsewhere. :)

4. Description of attack: Another nmap scan, no doubt, looking for open udp ports. Only thing that stands out is that this web server is on a completely different IP network, so its unlikely that the attacker was simply doing a sequential scan. They knew what they were looking for.

5. Attack mechanism: Reconnaissance. Look for open ports, and services listening on them, and attempt to find exploitable services/systems.

6. Correlations: Standard scan. Extremely common.

7. Evidence of active targeting: As mentioned in section 1. of this detect, this attacker is actively targeting our network. There is no numerical relation between the IP addresses of the systems they scanned earlier, and the one shown in this trace.

8. Severity: (4 + 2) - (4 + 5) = -3
   The target host is well secured. The only traffic allowed through the single point of entry is on ports 80 and 443.

9. Defensive recommendation: Defenses seem adequate, but why is this person so interested in us?

10. Multiple choice question: Which of the following is a commonly exploited UDP-based service?
    a) NFS
    b) FTP
    c) HTTP
    d) NNTP
    Answer: NFS

Detect 10:

[Sun Mar 19 11:51:55 2000] [error] [client 128.175.13.74] script not found or unable to stat: /var/www/data/cgi-bin/counterfiglet
[Sun Mar 19 21:55:25 2000] [error] [client 128.175.13.74] file permissions deny server execution: /var/www/data/cgi-bin/test-cgi
[Mon Mar 20 00:15:33 2000] [error] [client 128.175.13.74] script not found or unable to stat: /var/www/data/cgi-bin/phf
[Mon Mar 20 01:10:03 2000] [error] [client 128.175.13.74] script not found or unable to stat: /var/www/data/cgi-bin/aglimpse
[Mon Mar 20 18:58:26 2000] [error] [client 128.175.13.74] script not found or unable to stat: /var/www/data/cgi-bin/perl
[Tue Mar 21 00:41:30 2000] [error] [client 128.175.13.74] script not found or unable to stat: /var/www/data/cgi-bin/sh
[Tue Mar 21 01:37:05 2000] [error] [client 128.175.13.74] script not found or unable to stat:

1. Source of trace: My network

2. Detect generated by: apache webserver log

3. Probability of spoofed source address: Nil. This is a cgi scan, and the attacker needs to see the results.

4. Description of attack: The attacker is scanning the web server for scripts or cgis that can be exploited to gain access to the server. The most interesting thing about this is the *extremely* slow rate. He makes two to three attempts per day, no more. Almost certainly an attempt to avoid detection.

5. Attack mechanism: Once a particular script or cgi is found, the attacker can attempt to execute it or exploit it in some fashion so as to execute arbitrary commands on the server.

6. Correlations: Probably one of the most common methods of attacking a web server.

7. Evidence of active targeting: Certain. The attacker is targeting a web server with web/cgi scans.

8. Severity: (4 + 4) - (4 + 3) = 1 Its a fairly important webserver. Patches are up to date, but the firewall can do nothing to defend against this type of attack. It does, however block any other type of access to the machine.

9. Defensive recommendation: Increased watchfulness. None of the cgis the attacker wants/needs are available, although we might want to permanenty remove that test-cgi instead of simply making it unusable (chmod 0 was the mechanism used, it appears).

10. Multiple choice question: Which of the following programming languages is invulnerable to CGI exploits?
    a) C
    b) Perl
    c) Visual Basic
    d) None of the above
    Answer: (d)