Global Information Assurance Certification Paper

*** Northcutt, this arrived a bit late for the IDIC, but Eric did a nice job and we decided to pass this one and use it as an example for the San Jose folk's practical. Pass ***

Eric Francis
Practical submission for GIAC certification

Detect 1

(Laurie is doing some detecting down in .edu land)
Abovenet Communications, Inc., San Jose CA, USA
Mar 31 11:27:43 dns1 snort[4415]: spp_portscan: PORTSCAN DETECTED from 208.185.54.22
Mar 31 11:27:49 dns1 snort[4415]: spp_portscan: portscan
status from 208.185.54.22: 14 connections across 1 hosts:
TCP(0), UDP(14)
Mar 31 11:27:55 dns1 snort[4415]: spp_portscan:
End of portscan from 208.185.54.22
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33465 UDP
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33466 UDP
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33467 UDP
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33468 UDP
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33469 UDP
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33470 UDP
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33471 UDP
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33472 UDP
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33473 UDP
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33474 UDP
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33475 UDP
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33476 UDP
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33477 UDP
Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33478 UDP

1. Source of trace

GIAC: http://www.sans.org/y2k/040100.htm

2. Detect was generated by:

Snort intrusion detection system.

Fields:

Mar 31 11:27:43 208.185.54.22:33161 -> a.b.c.34:33478 UDP
| Date | Time | Source IP |Src Port| Dest IP |Dst Port|Protocol|

3. Probability the source address was spoofed

Unlikely. The attacker would needs responses to successfully determine the route.

4. Description of attack:

Portscan of traceroute ports. It's an automated tool, since we see 14 packets in 1 second and the source port is constant. A similar scan by the same source IP was performed on April 3 against a different set of ports. This scan had the same IP and source port as the first.

According to a reverse lookup, this IP is owned by Above.net. It also appears to be part of Speedera.com. During my research, I discovered that Above.net has a traceroute CGI script (http://www.above.net/cgi-bin/trace). Based on their website, they provide web hosting. I presume that this script is made available to their customers.

Based on these results, I suspect the detect is from a site hosted by Above.net that is attempting to perform load balancing.

5. Attack mechanism:

It is not an attack. It appears to be a load balancing attempt by a web server. I question the value of having a CGI script that permits anyone to enter an IP address for a traceroute.

6. Correlations:

Reverse lookup reveals:
Abovenet Communications, Inc. (NETBLK-ABOVENET-6)
50 W. San Fernando St., Suite 1010
San Jose, CA 95113

From: http://www.sans.org/y2k/040500-1230.htm

> (Today we have a special Laurie .edu edition with a most provocative top story ... my common sense said don't post this, but here we go anyway. When you make a detect and report it, who responds, who doesn't? Laurie has been keeping score. I found this to be fascinating reading.)
> I started doing the contacting myself with a Cc: to Randy. Here is a summary of the past month-ish as of this morning.
> Mar 31, 2000 208.185.54.22
> Automated response
> Response ("The following ip belongs to speedera.com. Is this still happening?")
> Answer ("It has happened again this morning [Apr 3]")

Later in the document:

> Abovenet Communications, Inc., San Jose CA, USA
> Apr 3 08:49:22 dns1 snort[4415]: spp_portscan:
> PORTSCAN DETECTED from 208.185.54.22
> Apr 3 08:49:28 dns1 snort[4415]: spp_portscan: portscan status
> from 208.185.54.22: 14 connections across 1 hosts: TCP(0), UDP(14)
> Apr 3 08:49:34 dns1 snort[4415]: spp_portscan: End of portscan
> from 208.185.54.22
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33512 UDP
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33513 UDP
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33514 UDP
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33515 UDP
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33516 UDP
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33517 UDP
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33518 UDP
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33519 UDP
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33520 UDP
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33521 UDP
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33522 UDP
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33523 UDP
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33524 UDP
> Apr 3 08:49:22 208.185.54.22:33161 -> a.b.c.34:33525 UDP

7. Evidence of active targeting:

Definitely active targeting, since it's coming from the same source IP and port on two occasions. Each scan is also looking at different destination ports.

8. Severity:

(3 + 0) - (4 + 4) = -5

Criticality = 3 (not sure what kind of destination)
Lethality = 0 (load balancing)
System Countermeasures = 4 (presuming good admin)
Network Countermeasures = 4 (no response indicated, running IDS)

9. Defensive recommendation:

Doesn't seem to be anything to worry about. I might contact Above.net again to determine if this is truly load balancing.

10. Multiple choice test question, write a question based on the trace and your analysis with your answer.

a) Load Balancing
b) RingZero
c) Port Scan
d) DNS buffer overflow
Answer: a

Detect 2

Apr 17 12:08:41 210.220.143.254:1533 -> z.y.x.34:1 SYN **S*****
Apr 17 12:08:41 210.220.143.254:864 -> z.y.x.34:111 SYN **S*****
Apr 17 12:08:42 210.220.143.254:865 -> z.y.x.34:111 UDP
Apr 17 12:08:43 210.220.143.254:1537 -> z.y.x.34:80 SYN **S*****
Apr 17 12:08:43 210.220.143.254:1539 -> z.y.x.34:79 SYN **S*****

1. Source of trace

GIAC: http://www.sans.org/y2k/041900.htm
One detect of many on Laurie's network that day.

2. Detect was generated by:

Looks like TCPdump.

Fields:
Apr 17 12:08:41 210.220.143.254:1533 -> z.y.x.34:1 SYN **S*****
| Date | Time | Source IP: Source Port |Dest IP:Port|Flags/Protocol|

3. Probability the source address was spoofed

Unlikely. Looks like a recon scan to determine the host type and services available. Would need the results to determine which attacks to run.

4. Description of attack:

They at performing a recon probe of this machine. First they try to TCP connect to port 1, to determine if it is an SGI Irix. Then it checks port 111 for SunRPC on both TCP and UDP. They then check to see if it has a web server (port 80). Finally they try to connect to finger on port 79.

5. Attack mechanism:

This is a recon probe. Based on the results of the probe, they could choose attacks to exploit potential vulnerabilities. First they checked for an SGI Irix, then a Sun. They finished up by checking for a web server and tried to get info via finger. I'd guess the scan was done by hand based on the times and limited number of ports scanned. The source port numbers show an interesting distribution. First we see one from port 1533 then ports 864 and 864. The scans from 864 and 865 were probably first, just got hung up in the ether. Based on the gaps between source ports, this isn't the only thing they are doing. There is no regular pattern of gaps between source ports, so they probably aren't probing other sites as part of an automated scan.

6. Correlations:

No other attacks from this IP have been noted. It's a fairly basic recon probe.

7. Evidence of active targeting:

Definitely active targeting since they involve a single source/destination IP pair with multiple ports of the destination scanned.

8. Severity:

(3 + 2) - (4 + 4) = -3

Criticality = 3 (not sure what kind of destination)
Lethality = 2 (worst case this machine is a Sun or SGI)
System Countermeasures = 4 (presuming good admin)
Network Countermeasures = 4 (no response indicated, running IDS)

9. Defensive recommendation:

It doesn't appear that the prober received any answers so they are still in the dark. I'd keep an eye open for them to come back. I'd also ensure that the probed system was locked down.

10. Multiple choice test question, write a question based on the trace and your analysis with your answer.

a) Port scan
b) Recon probe
c) DNS Version Scan
d) Syn flood
Answer: b

Detect 3

Road Runner Group, Herndon VA, USA DNS probe

```
Apr 11 05:32:59 hosth snort[87556]: spp_portscan:
PORTSCAN DETECTED from 24.27.209.180
Apr 11 05:33:05 hosth snort[87556]: spp_portscan:
portscan status from 24.27.209.180: 18 connections
across 18 hosts: TCP(18), UDP(0)
Apr 11 05:33:12 hosth snort[87556]: spp_portscan:
portscan status from 24.27.209.180: 35 connections
across 35 hosts: TCP(35), UDP(0)
Apr 11 05:33:18 hosth snort[87556]: spp_portscan:
End of portscan from 24.27.209.180
--------
Apr 11 05:32:59 24.27.209.180:1482 -> a.b.c.19:53 SYN **S*****
Apr 11 05:32:59 24.27.209.180:1489 -> a.b.c.26:53 SYN **S*****
Apr 11 05:32:59 24.27.209.180:1496 -> a.b.c.33:53 SYN **S*****
Apr 11 05:32:59 24.27.209.180:1514 -> a.b.c.51:53 SYN **S*****
Apr 11 05:32:59 24.27.209.180:1525 -> a.b.c.62:53 SYN **S*****
Apr 11 05:33:02 24.27.209.180:1546 -> a.b.c.83:53 SYN **S*****
Apr 11 05:32:59 24.27.209.180:1684 -> a.b.c.221:53 SYN **S*****
Apr 11 05:32:59 24.27.209.180:1695 -> a.b.c.232:53 SYN **S*****
Apr 11 05:32:59 24.27.209.180:1698 -> a.b.c.235:53 SYN **S*****
Apr 11 05:33:02 24.27.209.180:1652 -> a.b.c.189:53 SYN **S*****
```

1. Source of trace

GIAC: http://www.sans.org/y2k/041300.htm

2. Detect was generated by:

Snort.

3. Probability the source address was spoofed

Unlikely. Need the results of the scan to determine vulnerabilities.

4. Description of attack:

Looking for DNS servers/DNS zone transfer. I suspect that there was some previous recon or research to determine which destination IPs to hit since there is no logical sequence to them, i.e they didn't scan all IPs. Would be curious to know if one of these actually were a DNS server.

5. Attack mechanism:

It's a probe to find a DNS server(s). Once they find a server, they can try to perform a DNS transfer to gather more information about the network. Or attempt another DNS related attack such as a buffer overflow.

6. Correlations:

There are no other reports of scans from this network (24.27.209.x).

7. Evidence of active targeting:

It's an active targeting. They tested specific IPs on the network instead of a range.

8. Severity:

(5 + 1) - (4 + 4) = -2

Criticality = 5 (worst case, one of theses is a DNS server)
Lethality = 1 (probe, not attack)
System Countermeasures = 4 (presuming good admin, patches up to date)
Network Countermeasures = 4 (no response indicated, running IDS)

9. Defensive recommendation:

Ensure that none of the destinations are running DNS. If so, make sure the patches are up to date. If DNS server exists in this range and patches are out of date (sys countermeasures = 1), the severity jumps to 1.

10. Multiple choice test question, write a question based on the trace and your analysis with your answer.

a) DNS Server Scan
b) DNS Inverse Query
c) Port Scan
d) DNS buffer overflow
Answer: a

Detect 4

(Binette @home, I have tried to break these up a bit or easier reading. Just a textbook case of life in cablemodem land)

Apr 15 23:38:23 cc1014244-a kernel: securityalert: tcp if=ef0 from 206.100.37.200:4917 to 24.3.21.199 on unserved port 8080
Apr 15 23:38:23 cc1014244-a kernel: securityalert: tcp if=ef0 from 206.100.37.200:4918 to 24.3.21.199 on unserved port 1080

Apr 16 05:58:15 cc1014244-a kernel: securityalert: tcp if=ef0 from 210.68.177.120:16147 to 24.3.21.199 on unserved port 98
Apr 16 11:24:13 cc1014244-a kernel: securityalert: tcp if=ef0 from 202.99.81.139:1961 to 24.3.21.199 on unserved port 8080
Apr 16 12:41:53 cc1014244-a kernel: securityalert: udp if=ef0 from 63.27.191.123:1130 to 24.3.21.199 on unserved port 137
Apr 16 14:36:32 cc1014244-a kernel: securityalert: tcp if=ef0 from 207.181.96.5:1101 to 24.3.21.199 on unserved port 27374
Apr 16 17:35:07 cc1014244-a kernel: securityalert: tcp if=ef0 from 166.90.27.249:3042 to 24.3.21.199 on unserved port 27374
Apr 16 17:53:06 cc1014244-a kernel: securityalert: tcp if=ef0 from 171.217.239.90:3050 to 24.3.21.199 on unserved port 2578

1. Source of trace

GIAC: http://www.sans.org/y2k/041900.htm

2. Detect was generated by:

Not really sure. Probably something Linux.

Fields:
Apr 16 14:36:32 cc1014244-a kernel: securityalert: tcp if=ef0 from
| Date/Time | kernel = Linux? | Warn Msg |Proto|Net Device |

207.181.96.5:1101 to 24.3.21.199 on unserved port 27374
|Src IP | Src Port | Dest IP | Dest Port (inactive) |

3. Probability the source address was spoofed

Unlikely. Need responses to determine vulnerabilities.

4. Description of attack:

The first set of scans against ports 8080 and 1080 are looking for proxies. This is a quick scan from a single source.

The next day we see several scans from various sources through out the day. First there's a LinuxConf scan (98), followed by another proxy check (8080), NetBios (137), SubSeven twice (27374) and finally port 2578 which I haven't been able to track down.

5. Attack mechanism:

If you had a proxy running and the attacker was able to exploit it, they could use your site to hide attacks to other destinations. The attacks would appear to come from your site. More information is available at http://www.sans.org/y2k/y2k/proxy.htm and http://www.sans.org/y2k/y2k/socks.htm .

Exploiting a LinuxConf vulnerability would enable an attacker to take over your machine. The LinuxConf utility runs as root, and provides a GUI interface for configuring a Linux system.

NetBios has several vulnerabilities that can be exploited.

SubSeven is a well known trojan.

6. Correlations:

These come from Binette who seems to have a continuous low level of scans.

7. Evidence of active targeting:

These appear to be random scans.

8. Severity:

(2 + 1) - (4 + 4) = -4

Criticality = 2 (home PC)
Lethality = 1 (probe, not attack)
System Countermeasures = 4 (presuming good admin)
Network Countermeasures = 4 (no response indicated, running IDS)

9. Defensive recommendation:

Everything seems OK. Recommend keeping virus scanner up to date and periodic scans to protect against trojan infection. Make sure LinuxConf is set up correctly, if running. Appears firewall blocked probes.

10. Multiple choice test question, write a question based on the trace and your analysis with your answer.

Apr 16 05:58:15 cc1014244-a kernel: securityalert: tcp if=ef0 from 210.68.177.120:16147 to 24.3.21.199 on unserved port 98

a) DNS Zone Transfer
b) DNS Inverse Query
c) LinuxConf probe
d) DNS buffer overflow
Answer: c

Detect 5

@Home Network, Redwood City CA, USA
c66613-a.saltlk1.ut.home.com
Imap
Apr 8 02:18:56 dns3 snort[5209]: spp_portscan:
PORTSCAN DETECTED from 24.8.159.30
Apr 8 02:18:56 dns3 snort[5209]: SCAN-SYN FIN: 24.8.159.30:0 ->
z.y.x.98:143
Apr 8 02:19:02 dns3 snort[5209]: spp_portscan: portscan status
from 24.8.159.30: 1 connections across 1 hosts: TCP(1), UDP(0) STEALTH
Apr 8 02:19:08 dns3 snort[5209]: spp_portscan: End of portscan
from 24.8.159.30

[**] SCAN-SYN FIN [**]
04/08-02:18:55.897992 24.8.159.30:0 -> z.y.x.98:143
TCP TTL:231 TOS:0x0 ID:47106
SF** Seq: 0x32030000 Ack: 0x0 Win: 0x200

Apr 8 02:18:56 24.8.159.30:0 -> z.y.x.98:143 SYNFIN **SF****

1. Source of trace

GIAC: http://www.sans.org/y2k/041200.htm

2. Detect was generated by:

Snort.

3. Probability the source address was spoofed

Unlikely. Although the packet is crafted (source port 0, Syn and Fin flags), the attacker would need a response to determine if the vulnerability existed.

4. Description of attack:

Attack against TCP port 143 IMAP. There are many well known IMAP vulnerabilities that exist under most operating systems. The purpose of the Syn-Fin flags being set is to either evade detection by a firewall or ID system an/or identify the destination as a Linux box (linux responds to Syn-Fin with a Syn-Fin-Ack). The source port of 0 as well as the impossible flag combination indicates a crafted packet.

5. Attack mechanism:

IMAP vulnerabilities are common. If the attacker were able to determine the IMAP version being run, they could attempt to exploit an older version. If they determined that the destination was a Linux box, they could target the destination for common Linux vulnerabilities.

6. Correlations:

There have been no other activity reported to GIAC by this network segment (24.8.159.x).

7. Evidence of active targeting:

There is only one scan from this IP detected, however based upon the uniqueness of a crafted packet, it is targeted.

8. Severity:

(3 + 2) - (4 + 4) = -3

Criticality = 3 (not sure what kind of destination)
Lethality = 2 (probe, not attack. Would be higher if dest is Linux or runs IMAP)
System Countermeasures = 4 (presuming good admin)
Network Countermeasures = 4 (no response indicated, running IDS)

9. Defensive recommendation:

Defenses seem good. Presuming the site is not Linux nor running IMAP, shouldn't see this person again. Otherwise, make sure patches are up to date.

10. Multiple choice test question, write a question based on the trace and your analysis with your answer.

a) IMAP
b) DNS Inverse Query
c) Covert Channel 4708
d) Portmap
Answer: a

Detect 6

Something new from 225
Apr 21 21:38:13 cc1014244-a kernel: securityalert: udp if=ef0 from 24.3.21.225:2176 to 24.3.21.199 on unserved port 22
Apr 21 21:39:21 cc1014244-a kernel: securityalert: udp if=ef0 from 24.3.21.225:2181 to 24.3.21.199 on unserved port 5632
Apr 21 21:39:22 cc1014244-a kernel: securityalert: udp if=ef0 from 24.3.21.225:2181 to 24.3.21.199 on unserved port 22
Apr 21 21:39:52 cc1014244-a kernel: securityalert: udp if=ef0 from 24.3.21.225:2187 to 24.3.21.199 on unserved port 5632
Apr 21 21:39:52 cc1014244-a kernel: securityalert: udp if=ef0 from 24.3.21.225:2187 to 24.3.21.199 on unserved port 22
Apr 21 21:41:04 cc1014244-a kernel: securityalert: udp if=ef0 from 24.3.21.225:2188 to 24.3.21.199 on unserved port 5632
Apr 21 21:41:04 cc1014244-a kernel: securityalert: udp if=ef0 from 24.3.21.225:2188 to 24.3.21.199 on unserved port 22

1. Source of trace

GIAC: http://www.sans.org/y2k/042500.htm

2. Detect was generated by:

Unsure, probably Linux based. Field analysis above.

3. Probability the source address was spoofed

Unlikely, need response to choose attack. Also, multiple scans from this IP.

4. Description of attack:

Automated scan looking for PCAnywhere on UDP ports 22 and 5632. PCAnywhere versions 7.5 and below use UDP port 22. Versions above 7.5 use UDP port 5632.

5. Attack mechanism:

Automated scanner looking for machines running PCAnywhere. This could include a trial version being used to find other machines running PCAnywhere. We see two quick scans of port 5632 then 22, then wait, then another scan. Based on the source ports, it looks like they are just picking on this IP. Not sure why we only see the port 22 scan on the first round. Maybe attacker decided to expand scope to include additional PCAnyhwere port.

PCAnywhere is an admin tool that permits remote control of the machine. A successful exploit would turn control of the target over to the attacker.

6. Correlations:

This detect came from Lee Binette. He's seen a lot of activity from this IP, apparently another cable modem on his @Home network segment. This same type of attack occurred from this IP on April 24th.

7. Evidence of active targeting:

This source has been pestering this IP for a long time. Definitely active targeting.

8. Severity:

(2 + 1) - (4 + 4) = -5

Criticality = 2 (not sure what kind of destination, presumably home PC/network)
Lethality = 1 (probably not running PCAnywhere, if so lethality could be 5)
System Countermeasures = 4 (presuming not running PCAnywhere)
Network Countermeasures = 4 (no response indicated, running IDS)

9. Defensive recommendation:

This guy's had enough fun. Gather up the logs and report him!

10. Multiple choice test question, write a question based on the trace and your analysis with your answer.

a) SecureShell scan
b) HD's Spiderscanner
c) IMAP scan
d) PCAnywhere scan
Answer: d

Detect 7

04/01-15:59:26.043293 158.94.234.51:1674 -> MY.NET.70.227:6346
TCP TTL:117 TOS:0x0 ID:27853 DF
SFR**U21 Seq: 0x97FCBA Ack: 0x1141D Win: 0x5018
TCP Options => EOL EOL Opt 80 (40): 579C BBE0 E44A 83B0 0EC3
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0E C3 ..

04/01-16:00:33.741385 158.94.234.51:230 -> MY.NET.70.227:1674
TCP TTL:117 TOS:0x0 ID:3310 DF
SF**** Seq: 0x18CA0098 Ack: 0x5C0B141D Win: 0x5018
TCP Options => EOL EOL Opt 163 (40): E9E3 DC07 D411 A275 0060
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000

04/01-16:04:40.716885 158.94.234.51:1674 -> MY.NET.70.227:6346
TCP TTL:117 TOS:0x0 ID:61266 DF
SFRP**1 Seq: 0x996CFA Ack: 0x141D Win: 0x5018
TCP Options => EOL EOL Opt 238 (26): 0AE5 E007 D411 9F79 0010
0000 0000 0000 0000 0000 0000 0000 EOL EOL EOL EOL EOL EOL EOL

EOL EOL EOL EOL EOL

04/01-16:06:18.182252 158.94.234.51:1674 -> MY.NET.70.227:6346
TCP TTL:117 TOS:0x0 ID:46459 DF
SF*P*U1 Seq: 0xC30099 Ack: 0xDBD9141E Win: 0x5018
06 8A 18 CA 00 C3 00 99 DB D9 14 1E 06 AB 50 18 ....P.
00 00 D3 0A 00 00 A0 15 49 6C C4 07 D4 11 9F 25 ....Il.....%
00 10 ..

04/01-16:10:45.964767 158.94.234.51:1674 -> MY.NET.70.227:6346
TCP TTL:117 TOS:0x0 ID:5343 DF
SFRPAU21 Seq: 0xDB009A Ack: 0x7786141E Win: 0x5018
39 FF 50 18 00 00 EC A2 00 00 7B 15 49 6C C4 07 9.P.......{.Il..
D4 11 9F 25 00 10 ...%..

1. Source of trace

GIAC: http://www.sans.org/y2k/040400-000.htm
I took a set from the same source IP.

2. Detect was generated by:

TCPDump or some variant.

3. Probability the source address was spoofed

Maybe. Depends on the objective. If looking to identify the OS, they need the return packets. If the objective is to crash the destination, they don't care. The packets are obviously crafted due to the invalid flag settings.

4. Description of attack:

Five crafted packets over a period of almost 11 minutes. It appear to be a xmastree type scan, using various invalid combinations of TCP flags (such as a SYN, FIN, RESET and ACK). The first three packets appear to have invalid TCP options.
Four of the packets came from port 1674 while the second is from port 230. I didn't locate any special significance of these ports. Destination port 6346 is the default port for Gnutella servers.

The attacker is using SYN-FIN-ACK and SYN-FIN-RESET-ACK packets with various combinations of the PUSH and URGENT flags.

In a successful TCP connection, the source sends a SYN packet to the destination. If the destination accepts the connection, it replies with a SYN-ACK packet. The destination then replies with an ACK of the destination's SYN. Here, due to the invalid combinations of flags, no connection is possible. Both FINs and RESETs are used to break a connection. Using them in combination with a SYN flag prevents the connection from taking place.

5. Attack mechanism:

There are two possible purposes for this attack. The first is by setting various combinations of the TCP flags, the attacker could determine the machine type/OS in use at the destination. The other purpose could be to crash the machine with invalid packets.

The attack does not appear to be a standard xmastree scan. The purpose of a xmastree scan is to determine the type of OS the destination is running by the destination's response to invalid TCP flags. In this case, the combination of flags would not permit identification of the destination. A destination would not respond to a packet with the RESET flag set. A source packet with ACK would receive a RESET back, regardless if the port was open or closed.

The unusual TCP options makes me conclude that this is probably an attempt to crash a system with a bogus packet.

6. Correlations:

This detect came from Andy at .edu. There have not been any other reports of attacks from 158.94.234.x on the GIAC site. This IP is registered to Middlesex Polytechnic in England. I did not see any other attacks against this particular IP on Andy's network in GIAC.

7. Evidence of active targeting:

Active targeting since the source sent several packets to the same destination with different combinations of flags and options.

8. Severity:

(3 + 1) - (4 + 4) = -4

Criticality = 3 (not sure what kind of destination)
Lethality = 1 (probably not going to succeed)
System Countermeasures = 4 (presuming the destination didn't crash)
Network Countermeasures = 4 (no response indicated, running IDS)

9. Defensive recommendation:

An unusual scan but not likely to be very useful to the attacker.

10. Multiple choice test question, write a question based on the trace and your analysis with your answer.

a) Xmastree scan
b) SYN flood
c) Portmapper
d) DNS buffer overflow
Answer: a

Detect 8

FWIN,2000/04/29,15:47:54 -5:00 GMT, 216.37.13.181:1975, MY.NET.138.21:1397,TCP
FWIN,2000/04/29,15:48:12 -5:00 GMT, 216.37.13.179:1975, MY.NET.138.21:1408,TCP
FWIN,2000/04/29,15:49:16 -5:00 GMT, 216.37.13.184:1975, MY.NET.138.21:1475,TCP
FWIN,2000/04/29,15:49:26 -5:00 GMT, 216.37.13.177:1975, MY.NET.138.21:1480,TCP
FWIN,2000/04/29,15:50:46 -5:00 GMT, 216.37.13.178:1975, MY.NET.138.21:1548,TCP

1. Source of trace

This is from my dialup PC at home.

2. Detect was generated by:

Zone Alarm 2.1.7 with TrueVector 2.1.10.

3. Probability the source address was spoofed

The address is not spoofed (based upon analysis).

4. Description of attack:

This appears to be a port scan of my machine with several source IPs attempting to connect to several different ports on my machine. All source machines share the first three octets in their IP addresses. The source port does not change although it tries several different destination ports. The timing appears random but rather slow. These are all TCP attempts.

5. Attack mechanism:

At first, I thought I was being port scanned. I did not find any special relevance to the destination ports. Based on further investigation (searched for "port 1975" in Deja.com), it is not an attack, but a result of running Go!Zilla. It is an attempted back channel communication between Go!Zilla

and Aureate.com. ZoneAlarm doesn't track outgoing traffic so I can't see what my machine may be sending to Aureate. In this case, the attempts from Aureate to my machine are being blocked.

6. Correlations:

The IP 216.37.13.178 is attributed to Aureate.com and One Call Communications, Inc. From the One Call website:
"One Call Communications is a national communications services provider based in Carmel, Indiana. With services ranging from advanced internet backbone services to ordinary voice long-distance, Indiana-based One Call provides technical business and educational organizations with the tools to compete in a rapidly changing market."

Aureate is a company that provides a method to support advertisement supported software. According to http://www.theregister.co.uk/000323-000027.html:
"The software does, however, collect non-identifiable user information, and it does install its advertising features in the background. The background installation certainly is a questionable practice; but personally-identifiable information has never been collected by any Aureate products, which are free to users because they are supported by advertising, just as free ISPs are."

I did not locate any other sites reporting connection attempts from port 1975 in GIAC.

7. Evidence of active targeting:

Active targeting, although brought on by myself.

8. Severity:

(2 + 1) - (4 + 4) = -5

Criticality = 2 (Home PC)
Lethality = 1 (Not malicious)
System Countermeasures = 4 (I keep OS and virus scanners up to date)
Network Countermeasures = 2 (blocked incoming, don't know what went out)

9. Defensive recommendation:

While this isn't an attack and doesn't present a threat to the PC, I don't care to have someone potentially gathering information behind my back. I decided to use a utility called OptOut from http://www.grc.com that removes the ad software. Note that an uninstall of the Go!Zilla product will **not** remove the ad software.

10. Multiple choice test question, write a question based on the trace and your analysis with your answer.

a) DNS Zone Transfer
b) Aureate advertisement software
c) Portscan
d) File transfer
Answer: b

Detect 9

Name: selek.CS.Uni-Magdeburg.De
Address: 141.44.25.23
Apr 25 13:53:56 141.44.25.23:59430 -> xxx.xxx.xxx.002:8080 SYN **S*****
Apr 25 13:53:56 141.44.25.23:59429 -> xxx.xxx.xxx.001:8080 SYN **S*****
Apr 25 13:53:56 141.44.25.23:59432 -> xxx.xxx.xxx.004:8080 SYN **S*****
Apr 25 13:53:56 141.44.25.23:59431 -> xxx.xxx.xxx.003:8080 SYN **S*****
Apr 25 13:53:57 141.44.25.23:59440 -> xxx.xxx.xxx.005:8080 SYN **S*****
Apr 25 13:53:57 141.44.25.23:59459 -> xxx.xxx.xxx.006:8080 SYN **S*****
Apr 25 13:53:57 141.44.25.23:59478 -> xxx.xxx.xxx.007:8080 SYN **S*****
Apr 25 13:53:57 141.44.25.23:59479 -> xxx.xxx.xxx.008:8080 SYN **S*****

Apr 25 13:48:52.447 xxxxxxx1 kernel: 226 IP packet dropped (141.44.25.23->xxx.xxx.xxx.xxx: Protocol=TCP[SYN] Port 59494->8080): Restricted Port: Protocol=TCP[SYN] Port 59494->8080 (received on interface xxx.xxx.xxx.xxx)

1. Source of trace

http://www.sans.org/y2k/042800.htm

2. Detect was generated by:

According to the submission, this is a combination of Snort and firewall logs. The first 8 lines appear to be from Snort, the last line from the firewall.

3. Probability the source address was spoofed

Not likely, they are trying to connect to a commonly used proxy port.

4. Description of attack:

Fast but crude attempt to find a proxy port. The source ports are interesting. First we see 4 sequential attempts to destination IPs 1-4, then a gap of 7 source ports. Later on we see gaps in the source ports of 18 twice with the final two scans having sequential port numbers.

5. Attack mechanism:

It appears to be an automated tool (they arrive too fast for a manual scan) that is only searching this destination. If the tool were scanning other networks, I would expect a fairly consistent pattern of source port number gaps. Since the first 4 scans have sequential ports, I think they are only targeting this site. The later gaps in port numbers probably means they are off doing other things on the internet.

I discussed the implications of exploiting a proxy back in detect 4.

6. Correlations:

This detect came from Darren Webb. he reported another proxy scan (port 81) on the same day from a different source IP. There have not been other reports of activity from the 141.44.25.x network on GIAC. Proxy scanning is a well known detect.

7. Evidence of active targeting:

Active targeting, scan of a portion of the network.

8. Severity:

(3 + 1) - (4 + 4) = -4

Criticality = 3 (presuming destinations aren't 'special')
Lethality = 1 (scanning)
System Countermeasures = 4 (presuming good admin)
Network Countermeasures = 4 (detected scan, no apparent responses)

9. Defensive recommendation:

Looks like the scans were blocked and/or the destinations aren't running proxies.

10. Multiple choice test question, write a question based on the trace and your analysis with your answer.

a) Squid Proxy
b) Queso
c) DNS Version Scan
d) SubSeven
Answer: a

Detect 10

(A lil hint of infowar!)

Thought I'd send this on, (working on my detects!)

Apr 17 00:33:28 fwall 16 deny: icmp from 137.68.110.178 to xxx.xxx.7.0 type Echo Request
Apr 17 00:33:28 fwall 15 deny: icmp from 137.68.110.178 to xxx.xxx.7.255 type Echo Request
Apr 17 01:34:38 fwall 16 deny: icmp from 137.68.110.178 to xxx.xxx.7.0 type Echo Request
Apr 17 01:34:38 fwall 15 deny: icmp from 137.68.110.178 to xxx.xxx.7.255 type Echo Request
178.110.68.137.in-addr.arpa. 36235 PTR sail.kaist.ac.kr.
;; AUTHORITY RECORDS:
68.137.IN-ADDR.ARPA. 511431 NS NS.kaist.ac.kr.
68.137.IN-ADDR.ARPA. 511431 NS NS.KAIST.KR.APAN.NET.
;; ADDITIONAL RECORDS:
NS.kaist.ac.kr. 14631 A 143.248.1.177
NS.KAIST.KR.APAN.NET. 165831 A 192.249.24.62

Two separate queries to my Class C trying to determine what hosts I might have responding for further probing. These probes might also be looking for registered addresses that do not respond and that can be used in future DDOS attacks. From Korea, they hit both the older BSD .0 network address and the later .255 network address. I've seen more probes from kr recently and I'm wondering if a ramp-up is in progress aimed at a massive DDOS. From my network perspective I'm not too worried, but considering what's been going on with the stock market, we may be prime for a demoralizing attack if major centers can be shut down during trading hours.

1. Source of trace
http://www.sans.org/y2k/042000.htm

2. Detect was generated by:
Looks like a firewall log.

3. Probability the source address was spoofed
Probably not. They are probably looking for a site that could be used for an ICMP flood or identify network addresses for detailed scanning. If they are trying to flood KAIST (the source IP), then the source IP would be spoofed.

4. Description of attack:
They are sending an echo request packet to the two possible network broadcast addresses of the destination address.

5. Attack mechanism:
There could be several purposes of the attack:

The most likely purpose is that they are looking for networks that permit ICMP echo requests to a broadcast address. If this address supports the broadcast response, this network could be used as part of a denial of service (DOS) attempt against another network. They would spoof the source IP to be that of the network they wish to attack and send the packet to the broadcast address. This would result in a large number of echo replies going to the victim. If the attacker is able to gather enough broadcast networks, they could conduct an effective DOS attack.

An alternative purpose is similar to the first, in that they are spoofing the KAIST address with the hope of using this network as a source of replies for a DOS attack against KAIST. That's unlikely since there is no report of a previous recon effort to determine if this network supports echo reply broadcasts, although they report seeing other scans from Korea. They don't specify if the other scans were from this IP.

Another purpose that would be related to the first is that they are attempting to map the network. By sending in the echo request packet to the broadcast address they may be hoping that all active

machines on the network would reply. By capturing the replies, the attacker could determine which IPs on the network are active. They could follow this up with a more in depth scan (such as portscans) against selected targets in the network.

If the destination network were to respond to the echo request, the responses could tie up a significant portion of the network bandwidth.

6. Correlations:

There are no other reports in the GIAC listings for this network (137.68.110.x).

7. Evidence of active targeting:

Active targeting. Two attempts and it's unlikely there's a benign reason to echo request broadcast addresses.

8. Severity:

(4 + 2) - (3 + 5) = -2

Criticality = 4
Lethality = 2 (scanning)
System Countermeasures = 3
Network Countermeasures = 5 (denied request)

9. Defensive recommendation:

The echo request was denied by the firewall.

10. Multiple choice test question, write a question based on the trace and your analysis with your answer.

a) DNS Zone Transfer
b) Smurf
c) Portmap
d) Broadcast echo request
Answer: d