Practical Submission - from David Goch

GIAC Intrusion Detection Practical Assignment 1 – Network Detects

Snort Detect 1 - DDoS - mstream handler to client

```
[**] CAN-2000-0138 – DDoS – mstream handler to client [**]
02/22-14:27:50.303014 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x88
216.129.xx.xxx:12754 -> 142.xx.xx.xx:443 TCP TTL:111 TOS:0x0 ID:7958 IpLen:20 DgmLen:122 DF

***AP*** Seq: 0x9C6B  Ack:  0xCF20CAEA  Win:  0x2058  TcpLen:  20
16 03 00 00 4D 01 00 00 49 03 00 3A 95 68 74 04          ….M…I..:.ht.
3F 1E 9D A1 1B 8B BB B1 73 B6 4B 92 DB 07 C9 33          ?…….s.K….3
8C 55 C1 E1 2A 51 CD 3C ED A6 F3 10 3E FE 50 9A          .U..*Q.<….>.P.
88 E2 0E D4 50 4D 0D 65 65 D6 27 5B 00 1200 04           ….PM.ee.'[….
FE FF 00 0A FE FE 00 09 00 64 00 62 00 03 00 06          ………d.b….

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

[**] CAN-2000-0138 – DDoS – mstream client to handler [**]
02/22-14:27:50.307214 0:D0:B7:73:17:37 -> 0:2:4A:F6:60:0 type:0x800 len:0x75
142.xx.xx.xx:443 -> 216.129.xx.xxx:12754 TCP TTL:128 TOS:0x0 ID:64208 IpLen:20 DgmLen:103 DF

***AP*** Seq: 0xCF20CAEA  Ack:  0x9CBD  Win:  0x2006  TcpLen:  20
16 03 00 00 3A 02 00 00 36 03 00 3A 95 67 EB 8C          ….:…6..:.9..
95 C3 B9 E4 CA 94 2E 4D A5 65 9A E6 D7 CF C4 4E          …….M.e…..N
2F FD 67 3B 81 1C 7D C1 82 0A 96 10 3E FE 50 9A          /.9;..}…..>.P.
88 E2 0E D4 50 4D 0D 65 65 D6 27 5B 00 0A 00             ….PM.ee.'[…

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

[**] CAN-2000-0138 – DDoS – mstream client to handler [**]
02/22-14:27:50.678901 0:D0:B7:73:17:37 -> 0:2:4A:F6:60:0 type:0x800 len:0x1C3
142.xx.xx.xxx:443 -> 216.129.xx.xxx:12754 TCP TTL:128 TOS:0x0 ID:64218 IpLen:20 DgmLen:437 DF

***AP*** Seq: 0xCF20CB74  Ack:  0x9EC5  Win:  0x1DFE  TcpLen:  20
17 03 00 01 88 C9 C4 FF 97 7D F8 86 7F 46 B9 96          ………}…F..
E1 97 A7 2D 56 C2 38 DF E0 1A 44 45 1E B9 01 BF          …-V.8…DE….
…
…
AO 73 5B 7D 3C 4B 87 A7 C9 E6 AE 18 6B                   .s[}<K……k

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

[**] CAN-2000-0138 – DDoS – mstream client to handler [**]
02/22-14:27:50.682285 0:D0:B7:73:17:37 -> 0:2:4A:F6:60:0 type:0x800 len:0x473
142.xx.xx.xx:443 -> 216.xxx.xx.210:12754 TCP TTL:128 TOS:0x0 ID:64218 IpLen:20 DgmLen:1125 DF

***AP*** Seq: 0xCF20CD01  Ack:  0x9EC5  Win:  0x1DFE  TcpLen:  20
17 03 00 04 38 21 46 27 2B E5 1D E4 0D 80 AF 66          ….8!F'+……f
27 1A 7D 51 F1 96 89 27 48 84 85 E8 D1 51 5E 57          '.}Q…'H….Q^W
…
```

```
…
71 0C F3 D7 94 DA 5E 1B 93 E6 61 A3 E9                    q…..^…a..
```

Snort Signature

```
alert tcp any any -> any 12754  (msg:  “CAN-2000-0138” – DDoS – mstream client to handler”;
content:”>”; flags: AP;)
alert tcp any any -> any 15104  (msg:  “CAN-2000-0138” – IDS111 – DDoS - mstream client to handler”;
flags: S;)
```

Source of trace

The detect is from our company’s Internet connection. The sensor is located outside the firewall which filters inbound and outbound Internet traffic.

Detect was Generated by

The Snort Intrusion Detection system 1.7 generated the alert. The alert is reported to the network operator through SnortSnarf v111500.1. The alert and datagram portion of the packet were logged to disk.

Probability the source address was spoofed

Low. The snort sensor is between our network and the Internet. The exchange involves a limited 2-way conversation between our host and a host located outside our network. The destination address exists onour network and does offer services on port 443.

I resolved the IP address of the other host. It belongs to one of our customers. A traceroute was done back to this IP address and the TTL values recorded are reasonable. This traffic can be categorized as friendly fire that has generated a false positive.

Description of attack

Two senarios are possible:

1. False Positive:

The server is a web based client server application. The client has chosen port 12754 as part of a normal internal selection process. The client is communicating as expected entering data and receiving responses and data.

Because the conversation link is encrypted via SSL (port 443), our hex data evidence gives no clue as to the nature of the data transferred. In addition though it does provide evidence that the mstream content signature portion is matching against an encrypted data pattern. This provides evidence in favor of a false positive.

A search of the ‘who is’ owner for the IP address indicates that the other party in the communication is a firm that we do business with. Our application owner indicated that the firm is one that has been given access to the Web application and is a valid user. An examination of the server logs indicates that valid access was being done at the time and date of the alert.

As mentioned in the Correlation paragraph below, I have seen no prior detects for this alert.

The server which runs a modern hardened operating system is located in our DMZ area.

Although I presently view this detect as a false positive, I have put this alert on my own watch list. If the detect reoccurs in the near future I will have to investigate further.

2. Mstream DDOS

The second scenario is that the alert is valid and a problem exists. Because the destination port and data content is associated with the SNORT mstream Distributed Denial of Service (DDOS) program signature, this suggests that our site has been compromised and contains a DDOS master, handler or agent program.

DDOS Master

Under this scenario our host would be compromised and the attacker is communicating as a master from our system to an mstream handler program on the foreign host. According to the CERT IN-2000-05 documentation, mstream master program would communicates with the handler via a TCP connection and the communication would involve the exchange of some basic commands and data. From the example described in the CERT documentation the number of bytes sent are less than 50. Our detect indicates that from 117 to 1139 bytes are exchanged in each packet. This makes it less likely that a master program is communicating with a handler program. In addition there is only one IP destination detected. If there were communications with a handler I would expect to see multiple handler IPs and associated traffic.

DDOS Handler

According to the CERT IN-2000-05 Incident Note, communications between handler and agent takes place using the UDP protocol. Snort is alerting on TCP packets. We know this from such information as the sequence numbers and AP flags. Evidence suggests that the transmission is not consistent with handler / agent communications.

Agent to Target

The amount of traffic entering or leaving our site is insufficient to indicate that an active DDoS attack is in progress.

Attack Mechanism

Snort associates the attack signature with reference number CAN-2000-0138. I looked up the reference number on cve.mitre.org, common vulnerabilities and exposures web site. A further reference (20000429) from this site to the BUGTRAQ web site could not be located on that site.

The CVE description indicates that a distributed denial of service (DDOS) attack master, handler, or agent program may have been installed on the host. Suspect programs include Trinoo, Tribe Flood Network (TFN / TFN2K), stacheldraht, mstream or shaft. The writeup goes on to indicate that the use of a host for attack programs is diminished for hosts implementing egress filtering. Egress filtering basically checks for outgoing traffic containing source IPs that are not consistent with the source network. When found such traffic, which usually indicates spoofing, should be blocked.

A network of DDOS hosts are characterized by a master host sending commands to handler or agent hosts that are commonly located on other compromised hosts in one or more countries. The agents then generate DOS traffic to the target host IP address in a form consistent with the initiating attack master's request. The agent hosts can use spoofed source IPs and generate different types of traffic such as echo requests, SYN floods, UDP floods, stream (ACK), etc. The requests can overwhelm the target host and cause the server to degrade service or crash.

Correlations

This is the first time I have noticed this alert. The other IP involved in the TCP/IP connection is a valid customer IP address. I have not noticed this customer's IP associated with problems in the past. The source address does not appear in the SANS database.

Evidence of Active Targeting

There is evidence of active targeting as the TCP / IP connection and exchange is between two specific hosts and port numbers.

Severity

Severity Measure = (Criticality + Lethality) – (System Countermeasures + Network Countermeasures)
= 5 – 8 = -3

This coefficient indicates that based on this analysis the detect appears to be an acceptable risk.

Criticality: 5, the host is an important gateway web server application to the rest of the critical web server farm.

Lethality: 0, as the detect is assumed a false positive.

System Countermeasures: 4, Host modern operating system is fully hardened with non-essential services disabled. Access is limited to console login.

Network Countermeasures: 4, Current firewall in place and separate IDS system operational.

Defensive Recommendation

As mentioned in the discussion on Attack Mechanism, egress filtering is a recommended defense that discourages someone from installing a DDOS attack trojan program. The reason is that it limits the desirability of the host as any outgoing spoofed source IP addresses from the program are detected and blocked by the perimeter firewall.

This detect was discussed with Firewall Support. I learned that the firewall was replaced with a new vendor's product 2 years ago. Prior to this egress filtering was performed. A follow-up with the current firewall vendor will be done by Firewall Support to learn whether this feature is automatically bundled as part of the base rule set in the current firewall.

The detect and followup analysis was reviewed with the system administrator responsible for the host involved. This was performed as an awareness exchange as well as a check on the integrity of the original analysis and main severity assessment.

An integrity check on the Snort rule is recommended. A query to the Snort site will be sent to reconfirm the validity of the detect. This is because CERT literature suggests that the port numbers are configurable. In addition, the default port numbers (6723, 6838, 7983 and 9325) documented by CERT do not match the port number (12754) in the Snort signature rule. This leads to less confidence in the validity of the signature.

Multiple choice test question

```
[**] CAN-2000-0138 – DDoS – mstream handler to client [**]
02/22-14:27:50.303014 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x88
216.129.xx.xxx:12754 -> 142.xxx.xx.xxx:443 TCP TTL:111 TOS:0x0 ID:7958 IpLen:20 DgmLen:122 DF

***AP*** Seq: 0x9C6B  Ack:  0xCF20CAEA  Win:  0x2058  TcpLen:  20
16 03 00 00 4D 01 00 00 49 03 00 3A 95 68 74 04          ….M…I..:.ht.
3F 1E 9D A1 1B 8B BB B1 73 B6 4B 92 DB 07 C9 33          ?…….s.K….3
8C 55 C1 E1 2A 51 CD 3C ED A6 F3 10 3E FE 50 9A          .U..*Q.<….>.P.
88 E2 0E D4 50 4D 0D 65 65 D6 27 5B 00 1200 04           ….PM.ee.’[….
FE FF 00 0A FE FE 00 09 00 64 00 62 00 03 00 06          ………d.b….
```

In the Snort alert listed above, the message ‘mstream handler to client’ is:

a) text from a common CB radio broadcast
b) text contained in the Snort program source code
c) text located in a Snort rule
d) text contained in the data portion of the packet

Answer: c)

Snort Detect 2 - SYNFIN Scan

```
Feb 22 13:20:07  141.223.165.221:21 -> 142.xxx.xxx.xxx:21  SYNFIN  ******SF

[**] IDS441-SCAN – Synscan Portscan [**]
02/22-13:20:07.987726 0:2:4A:F6:60:0 -> 0:C0:4F:BF:9:C9 type:0x800 len:0x3C
141.223.165.221:21 -> 142.xxx.xxx.xxx:21 TCP TTL:25 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x6509223E Ack: 0x267903A9 Win: 0x404 TcpLen: 20

Feb 22 13:20:08  141.223.165.221:21 -> 142.9.xx.xx:21  SYNFIN  ******SF

[**] IDS441-SCAN – Synscan Portscan [**]
02/22-13:20:08.146647 0:2:4A:F6:60:0 -> 0:0:A9:6:47:7E type:0x800 len:0x3C
141.223.165.221:21 -> 142.xxx.xxx.xxx:21 TCP TTL:25 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x6509223E Ack: 0x267903A9 Win: 0x404 TcpLen: 20

Feb 22 13:20:18  141.223.165.221:21 -> 142.9.xx.xx:21  SYNFIN  ******SF

[**] IDS441-SCAN – Synscan Portscan [**]
02/22-13:20:18.166689 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
141.223.165.221:21 -> 142.xxx.xxx.xxx:21 TCP TTL:25 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x2FA24B42 Ack: 0x77195223 Win: 0x404 TcpLen: 20

Feb 22 13:20:23  141.223.165.221:21 -> 142.9.xx.xx:21  SYNFIN  ******SF

[**] IDS441-SCAN – Synscan Portscan [**]
02/22-13:20:23.927275 0:2:4A:F6:60:0 -> 0:0:52:25:2:E3 type:0x800 len:0x3C
```

```
141.223.165.221:21 -> 142.xxx.xxx.xxx:21 TCP TTL:25 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x42C1189E Ack: 0x746C71B0 Win: 0x404 TcpLen: 20
```

Source of trace

The detect is from our company's Internet connection. The sensor is located outside the firewall which filters inbound and outbound Internet traffic.

Detect was Generated by

The Snort Intrusion Detection system 1.7 generated the alert. The alert is reported to the network operator through SnortSnarf v111500.1. The alerts were logged to disk.

Probability the source address was spoofed

Low. The nature of the SF scan is such that nothing is learned if the response goes to someone else (ie. a spoofed source address).

Sometimes a scanner will generate additional scanning packets with spoofed source IP addresses and attempt hide in the increased traffic. A review of the detects indicated only one scan was detected against our destination IP addresses on that day.

The source IP address is 141.223.165.221. This resolves to Pohang Institute of Science and Technology – Computer Center, Korea. Universities and colleges are environments where students get free access to computers for learning and unfortunately some use this open environment to try out different reconnaissance and exploit tools. Foreign laws against reconnaissance and exploits are evolving but currently have not had enough success to discourage undesirable intrusion activity.

This source host may also have been compromised as operating system software associated with education schools is often not fully patched.

I submitted a tracert command to the source IP and received back a hop count of slightly in excess of 25 before a no response messages started appearing. The detect shows TTL values of 25 which indicates that this source PC is consistent with several OS including MAC OS and Sun OS which have default initial TTL values of 60. If the TTL values were significantly different than any of the default TTL values listed at http://www.map2.ethz.ch/ftp-probleme.htm then this would indicate crafting to me which would also increase the possibility of IP source address crafting as well.

Description of Attack

A series of SYNFIN packets was sent to a limited number of hosts on our class B network.

The scan targeted the FTP port 21. The packets are normal looking except for the SF flags being set and the non zero acknowledgement numbers.

The length fields in the packet indicate normal IP header and TCP header lengths of 20 bytes. The IP datagram length values of 40 indicate that no data is sent.

A window size TCP option of 1028 bytes is specified in every alternate SF packet sent. The default size is not consistent with a Windows (8760 bytes) or Linux host (32120 bytes).

The short timeframe between the two slightly different packets sent at each host indicates that a program is perhaps being used to generate the packets.

The 9 nmap program OS tests as documented in the SANS material do not include SF packets among their 9 tests. Most other OS fingerprinting scan tools appear to send a stream of different tests with varying field contents. The detect is capturing fewer packets than I would expect if an OS were being fingerprinted and the packets are not targeting many different hosts as I would expect from a more general reconnaissance exercise.

One possibility is that someone may be in the process of creating a new scan program variant. The limited scan is consistent with what I would expect to see from a developer testing a program. This is also consistent with what I would expect from computer student learning about reconnaissance program methodology.

Attack Mechanism

The SYNFIN (SF) scan was originally developed to evade earlier IDS scanners which had their signatures set to detect TCP SYN (S) packets. Some older router based firewall systems also tend to be more forgiving in allowing in packets with FIN flags. The FIN packet is more indicative of a previously established connection.

The SF packet scan is normally a noisy reconnaissance technique that is easily detected by present releases of IDS signatures.

The reconnaissance technique involves sending a TCP SF flag combination packet to a host and observing the response. The normal response to a TCP SYNFIN (SF) packet from a listening host is a TCP RESET (R) packet sent back to the source host.

If the destination host doesn't exist or is temporarily unavailable, the router will return a ICMP Host Unreachable message. In addition as in this case the attacker can target a specific service (eg. FTP – port 21). If the host exists but not listening on the port, an ICMP Port Unreachable message is returned. A scan against port 21 is a good approach because FTP to the Internet is common and frequently allowed. This increases the success of an attacker receiving a response.

The attacker receives an indication of which IP addresses have live hosts and which live hosts have a listening FTP service on port 21.

Based on the responses received, the attacker can then target a host to determine the operating system. From this information appropriate exploits can be selected and run against the target.

Correlations

Our IDS history database is a relatively recent addition to our IDS toolkit. I searched our history data and noted that this is the first time this source IP and related subnet has been detected by our Snort Intrusion Detection System. The IP address is also not listed on the SANS web site Consensus Intrusion Database (Top Ten Source IPs Detected List).

Evidence of Active Targeting

There is some evidence of active targeting.

The scan is a small scan directed at only 4 different hosts. The scan could have stepped through all the hosts on our subnet but for reasons unknown was only against these four host IP addresses.

Since each of the hosts scanned represents a live host there must have been some earlier reconnaissance performed. There is no previous history of alerts from this source IP, so if previous attacks were conducted they either:

- used spoofed IP addresses
- were conducted through another computer using an ISP from a different IP subnet
- were carried out using a stealthy scan that wasn't detected by our IDS
- used an IDS insertion / evasion technique.

Severity

Severity Measure = (Criticality + Lethality) – (System Countermeasures + Network Countermeasures)
= 5 – 4 = 1

This coefficient indicates that based on this analysis the scan appears to be an acceptable risk.

Criticality: 3: The hosts run miscellaneous infrastructure services that support general datacentre Information Services Operations.

Lethality: 2: None of the servers scanned offer an FTP service and are not listening on the FTP port. As mentioned in the paragraph titled Description of Attack one plausible motive for the scan is to test a program and not to compromise a specific host.

System Countermeasures: 4: Our FTP hosts have fully hardened modern operating systems with non-essential services disabled. Access to our FTP service requires a valid userid and password. The use of a Secureid time based remote access token is required as part of the password entered when logging in.

Network Countermeasures: 0: Access to port 21 is allowed through the firewall. Therefore network countermeasures are not available.

Defensive Recommendation

An email will be sent to the source IP contact commenting on the scan. Future activity from this source IP will be monitored. The Source IP will be added to shunning list (router ACL deny rule) if the attacker persists.

Multiple choice test question

```
Feb 22 13:20:07  141.223.165.221:21 -> 142.xxx.xxx.xxx:21  SYNFIN  ******SF
```

```
[**] IDS441-SCAN – Synscan Portscan [**]
02/22-13:20:07.987726 0:2:4A:F6:60:0 -> 0:C0:4F:BF:9:C9 type:0x800 len:0x3C
141.223.165.221:21 -> 142.xxx.xxx.xxx:21 TCP TTL:25 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x6509223E Ack: 0x267903A9 Win: 0x404 TcpLen: 20
```

```
Feb 22 13:20:08  141.223.165.221:21 -> 142.9.xx.xx:21  SYNFIN  ******SF
```

A SYNFIN scan such as pictured above is used by an attacker to:

a) obtain reconnaissance information to determine live hosts and ports
b) obtain reconnaissance information in order to fingerprint the host's operating system
c) all of the above
d) none of the above

Answer: a)

Snort Detect 3 - Possible Queso Fingerprint attempt

```
[**] IDS29 – SCAN – Possible Queso Fingerprint attempt [**]
02/23-02:18:50.087900 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x42
64.152.66.27:35504 -> 142.xxx.xxx.xxx:53 TCP TTL:50 TOS:0x0 ID:0 IpLen:20 DgmLen:52 DF
12****S* Seq: 0x41B2F58D Ack: 0x0 Win: 0x16D0 TcpLen: 32
TCP Options (6) => MSS: 1460 NOP NOP SackOK NOP WS: 0

[**] IDS29 – SCAN – Possible Queso Fingerprint attempt [**]
02/23-02:18:52.086742 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x42
64.152.66.27:35530 -> 142.xxx.xxx.xxx:53 TCP TTL:50 TOS:0x0 ID:0 IpLen:20 DgmLen:52 DF
12****S* Seq: 0x427604FD Ack: 0x0 Win: 0x16D0 TcpLen: 32
TCP Options (6) => MSS: 1460 NOP NOP SackOK NOP WS: 0

Feb 23 02:18:54 64.152.66.27:35553 -> 142.9.3.3:53 SYN 12****S* RESERVEDBITS

[**] IDS29 – SCAN – Possible Queso Fingerprint attempt [**]
02/23-02:18:54.085996 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x42
64.152.66.27:35553 -> 142.xxx.xxx.xxx:53 TCP TTL:50 TOS:0x0 ID:0 IpLen:20 DgmLen:52 DF
12****S* Seq: 0x41BE01C3 Ack: 0x0 Win: 0x16D0 TcpLen: 32
TCP Options (6) => MSS: 1460 NOP NOP SackOK NOP WS: 0

[**] IDS29 – SCAN – Possible Queso Fingerprint attempt [**]
02/23-15:06:46.096066 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x42
64.152.66.27:42511 -> 142.xxx.xxx.xxx:53 TCP TTL:51 TOS:0x0 ID:0 IpLen:20 DgmLen:52 DF
12****S* Seq: 0x9612F3AF Ack: 0x0 Win: 0x16D0 TcpLen: 32
TCP Options (6) => MSS: 1460 NOP NOP SackOK NOP WS: 0

[**] IDS29 – SCAN – Possible Queso Fingerprint attempt [**]
02/23-15:06:48.096549 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x42
64.152.66.27:42614 -> 142.xxx.xxx.xxx:53 TCP TTL:51 TOS:0x0 ID:0 IpLen:20 DgmLen:52 DF
12****S* Seq: 0x964C51AA Ack: 0x0 Win: 0x16D0 TcpLen: 32
TCP Options (6) => MSS: 1460 NOP NOP SackOK NOP WS: 0

Feb 23 15:06:50 64.152.66.27:42726 -> 142.9.3.3:53 SYN 12****S* RESERVEDBITS

 [**] IDS29 – SCAN – Possible Queso Fingerprint attempt [**]
02/23-15:06:50.094164 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x42
64.152.66.27:42726 -> 142.xxx.xxx.xxx:53 TCP TTL:51 TOS:0x0 ID:0 IpLen:20 DgmLen:52 DF
12****S* Seq: 0x9710FD05 Ack: 0x0 Win: 0x16D0 TcpLen: 32
TCP Options (6) => MSS: 1460 NOP NOP SackOK NOP WS: 0

[**] BETA – IDS162 – PING Nmap2.36BETA or HPING2 Echo from LINUX/*BSD [**]
02/23-15:19:13.344154 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
```

```
64.152.66.27 -> 142.xxx.xxx.xxx ICMP TTL:51 TOS:0x0 ID:0 IpLen:20 DgmLen:28 DF
Type:8 Code:0 ID:45659 Seq:256 ECHO

[**] ICMP Unknown Type [**]
02/23-15:19:13.344789 0:D0:B7:73:17:37-> 0:2:4A:F6:60:0 type:0x800 len:0x3C
142.xxx.xxx.xxx -> 64.152.66.27 ICMP TTL:255 TOS:0x0 ID:36874 IpLen:20 DgmLen:28
Type:0 Code:0 ID:45659 Seq:256 ECHO REPLY


[**] BETA – IDS162 – PING Nmap2.36BETA or HPING2 Echo from LINUX/*BSD [**]
02/23-15:19:13.427713 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
64.152.66.27 -> 142.xxx.xxx.xxx ICMP TTL:51 TOS:0x0 ID:0 IpLen:20 DgmLen:28 DF
Type:8 Code:0 ID:45659 Seq:256 ECHO

[**] ICMP Unknown Type [**]
02/23-15:19:13.428262 0:D0:B7:73:17:37-> 0:2:4A:F6:60:0 type:0x800 len:0x3C
142.xxx.xxx.xxx -> 64.152.66.27 ICMP TTL:255 TOS:0x0 ID:36875 IpLen:20 DgmLen:28
Type:0 Code:0 ID:45659 Seq:256 ECHO REPLY

 [**] BETA – IDS162 – PING Nmap2.36BETA or HPING2 Echo from LINUX/*BSD [**]
02/23-15:19:13.513320 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
64.152.66.27 -> 142.xxx.xxx.xxx ICMP TTL:51 TOS:0x0 ID:0 IpLen:20 DgmLen:28 DF
Type:8 Code:0 ID:45659 Seq:256 ECHO

[**] ICMP Unknown Type [**]
02/23-15:19:13.513867 0:D0:B7:73:17:37-> 0:2:4A:F6:60:0 type:0x800 len:0x3C
142.xxx.xxx.xxx -> 64.152.66.27 ICMP TTL:255 TOS:0x0 ID:36874 IpLen:20 DgmLen:28
Type:0 Code:0 ID:45659 Seq:256 ECHO REPLY
```

Snort Signature

```
alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"IDS29 - SCAN-Possible Queso F
ingerprint attempt";flags:S12;)

alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"BETA - IDS162 - PING Nmap2.3
6BETA or HPING2 Echo from LINUX/*BSD";itype:8;dsize:0;)

alert icmp any any -> any any (msg:"ICMP Unknown Type";)
```

Source of trace

The detect is from our company's Internet connection. The sensor is located outside the firewall which filters inbound and outbound Internet traffic.

Detect was Generated by

The Snort Intrusion Detection system 1.7 generated the alert. The alert is reported to the network operator through SnortSnarf v111500.1. The alerts were logged to disk.

Probability the source address was spoofed

Low. The snort sensor is between our network and the Internet. The nature of a fingerprinting reconnaissance scan is that the originator must receive back the results of the scan in order to predict the OS. Other scans of this nature were not received on this day.

The trace route against the scanning host IP address returned a value of 17 hops. Adding to the TTL value shown on the detect gives 68. This does not seem to be a reasonable value and suggests that the normal initial TTL value may have altered. Since 17 hops occurred back to the source the attacker may have increased the normal initial default TTL value for his OS. Attack tools provide the flexibility to alter parameters.

Description of attack

Two senarios are possible:

1. False Positive - Bot:

A client has visited a web site associated with the source IP address and the site has launched a bot to determine the best route in order to communicate with our host.

A search of the 'who is' owner for the source IP address indicates that the owner of the IP is Level 3 Communications based in Colorado. The IP addresses are assigned to www.webzone.net. A visit to the web site indicated that their business is B2B, web hosting and they advertise a 'Yipes that's fast' slogan.

It is common for performance software such as 3DNS to attempt to assign regional web servers to customers by sending different forms of traffic against the customer's DNS server and analyzing the responses returned. The programs are also configured to send out one set of traffic and if no response is returned to automatically send another type of traffic.

Our sensor has recorded a series of 3 TCP SYN packets, and a wait time of 13 seconds which is consistent with the program waiting for RESET packets. This is followed by another attempt at sending the same 3 TCP SYN packets. After another period of 13 seconds a series of 3 echo requests is tried. This indicates that our DNS operating system may not have responded to the TCP reserved bit SYN packet.

2. Queso Fingerprint Attempt

The second scenario is that the alert is valid and a problem exists. The Queso signature is set to alert when a SYN packet having reserved bits set is observed. I located documentation on the queso program from www.securityfocus.com. The program sends out seven TCP IP packets at a time. Most of the packets with the exception of the SYN packet with the reserved bits set would not trigger an alert.

On examining the recorded traffic from SNORT, we see that there is about 2 seconds of delay between each of these SYN reserved bit packets. One reason for the delay is that the program is sending the other 6 types of packets to our host.

The followup echo requests however is not characteristic of queso.

Attack Mechanism

Snort associates the attack signature with reference number IDS29. This is a www.whitehats.com signature that has a link to CAN-1999-0454. I looked up the reference number and description on cve.mitre.org, common vulnerabilities and exposures web site. I was able to locate documentation on www.securityfocus.com describing the attack mechanism.

Queso is similar to NMAP in that it sends a series of packets and based on the responses returned attempts to identify the operating system. Once the attacker learns the operating system, known vulnerabilities can be researched and compromise attacks can be launched.

Correlations

This is the first time I have seen this Snort alert. Unfortunately we do not store network traffic in order to support the type of analysis that could confirm whether the additional documented queso signature packets were sent (see recommendations).

We have recently started to maintain an alert database by source IP. A search of this database did not provide a match.

Evidence of Active Targeting

There is evidence of active targeting to our DNS server. The DNS server is accessible from the Internet for DNS resolution. The server is also a valuable prize for a hacker to control. Web site redirection to other IP addresses is possible if a DNS server becomes compromised.

Severity

Severity Measure = (Criticality + Lethality) – (System Countermeasures + Network Countermeasures)
= 7 – 8 = -1

This coefficient indicates that based on this analysis the detect appears to be an acceptable risk.

Criticality: 5, the host is our DNS server which is the authority DNS for our web site.

Lethality: 2, The detect is assumed to be a load balancing program but without a data warehouse of recent network traffic to see the full spectrum of communication I can't be sure.

System Countermeasures: 4, Host modern operating system is fully hardened with non-essential services disabled.

Network Countermeasures: 4, The DNS is accessible to the Internet by necessity. Current firewall rules block TCP DNS connections from the Internet.

Defensive Recommendation

The implementation of our IDS system is in progress to ensure full coverage and that an adequate history of alerts is available for analysis. Although not implemented yet the implementation will incorporate the retention of all network traffic that occurs over a specified timeframe. This will allow us to perform more in depth analysis of all traffic and data that occurs during a conversation with an IP address.

I have sent a friendly email to the Technical Support staff at www.thewebzone.net. Included in the email is a description of the traffic observed and a request for them to verify if a load balancing performance program was involved.

Multiple choice test question

```
[**] IDS29 – SCAN – Possible Queso Fingerprint attempt [**]
02/23-15:06:50.094164 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x42
64.152.66.27:42726 -> 142.xxx.xxx.xxx:53 TCP TTL:51 TOS:0x0 ID:0 IpLen:20 DgmLen:52 DF
12****S* Seq: 0x9710FD05 Ack: 0x0 Win: 0x16D0 TcpLen: 32
TCP Options (6) => MSS: 1460 NOP NOP SackOK NOP WS: 0
```

Which of the following is not true:

a) Queso is an attack program used to fingerprint a host's operating system.
b) Queso is similar to NMAP.
c) Queso sends out a series of 7 TCP/IP packets.
d) Queso is a virus program.

Answer: d)

Snort Detect 4 - Possible attempt at MS Print Services

```
[**] OVERFLOW - Possible attempt at MS Print Services [**]
02/24-09:13:10.871674 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x4A
200.206.162.176:4771-> 142.9.xxx.xxx:515 TCP TTL:49 TOS:0x0 ID:20728 IpLen:20 DgmLen:60 DF
******S* Seq: 0xEA16CEF5 Ack: 0x0 Win: 0x7D78 TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 14090605 0 NOP WS: 0

[**] OVERFLOW - Possible attempt at MS Print Services [**]
02/24-09:13:10.878228 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x4A
200.206.162.176:4803-> 142.9.xxx.xxx:515 TCP TTL:49 TOS:0x0 ID:20760 IpLen:20 DgmLen:60 DF
******S* Seq: 0xEA3E9A3D Ack: 0x0 Win: 0x7D78 TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 14090606 0 NOP WS: 0

[**] OVERFLOW - Possible attempt at MS Print Services [**]
02/24-09:13:10.883146 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x4A
200.206.162.176:4834-> 142.xxx.xxx.xxx:515 TCP TTL:49 TOS:0x0 ID:20791 IpLen:20 DgmLen:60 DF
******S* Seq: 0xE9E03A13 Ack: 0x0 Win: 0x7D78 TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 14090606 0 NOP WS: 0

….etc.
```

Snort Signature

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 515 (msg:"OVERFLOW – Possible attempt at MS
Print Services";)
```

Source of trace

The detect is from our company's Internet connection. The sensor is located outside the firewall which filters inbound and outbound Internet traffic.

Detect was Generated by

The Snort Intrusion Detection system 1.7 generated the alert. The alert is reported to the network operator through SnortSnarf v111500.1. The alerts were logged to disk.

Probability the source address was spoofed

The snort sensor is between our network and the Internet. Our class B network was scanned for listening Microsoft print services. Systematic TCP SYN packets were sent to each host on print server port 515. The packets include certain TCP options.

There appears to be low probability of source IP spoofing.

The attack appears to be an attempt at discovering listening print servers. Hosts listening on port 515 will return a SYN ACK packet. Hosts not listening return an ICMP port unreachable message. Non existent hosts return ICMP Host Unreachable response.
The sequence numbers and source ports appear to be increasing as expected from a real host.

The source of the scan must receive the reconnaissance information for it to be of value. The source IP resolved to the Brazilian Research Network. This is a source of frequent malicious activity. The host may be compromised.

Description of Attack Scenarios

This appears to be a reconnaissance scan for listening printer daemon services.

A subnet host scan was performed to discover live hosts with listening print server ports.

The packets do not appear to be a DOS service due to malformed packets. The TCP options include a Window scale factor of 0, a timestamp of from 14090605 to 14090705, selective acknowledgement enabled, and an ethernet minimum segment size. A window size of 32,120 bytes (0x7D78) is consistent with a Linux OS. The TCP options appear reasonable and should not cause a buffer overflow or similar exploit.

There is a CERT Advisory (CA-2000-22) documented that describes a buffer overflow scenario using an snprintf() function call against TCP port 515. This may be a prelude to the execution of this exploit.

Attack Mechanism

The packet is sent and a response or lack of response is noted. Hosts listening on port 515 will return a SYN ACK packet. Hosts not listening return an ICMP port unreachable message. Non existent hosts return ICMP Host Unreachable response. The success of the reconnaissance also depends on firewall and router filtering rules. If the routers have been configured for 'no IP unreachables' than these messages are not sent back.

Correlations

I am starting to see more of these MS Print Service connection attempts. The attempts are coming from suspect source IP addresses from Asia and Brazil. This gives me an uneasy feeling as attacks frequently originate from these overseas sites. They can be a haven for hackers either from laxness in pursing attackers or through their existence as compromised hosts the attackers can hide in. As mentioned earlier, this may be a prelude to the execution of a buffer overflow attempt once an open port is found. This exploit is documented in CERT Advisory CA-2000-22.

Evidence of Active Targeting

There is evidence of active searching for listening print server ports. The source host may be analyzing the received responses. He may be mapping hosts or if he is launching a well-planned exploit, noting how the hosts that are exposed to the exploit react. If responses are an important part of the attack then compromised hosts are an important vehicle from which to observe the response.

Severity

Severity Measure = (Criticality + Lethality) – (System Countermeasures + Network Countermeasures)
= 7 – 6 = 1

This coefficient indicates that based on this analysis the detect appears to be an acceptable risk.

Criticality: 3, The firewall blocks access to port 515 from the internet. Our hosts are not susceptible to buffer overflow attempts as documented in CERT CA-2000-22.

Lethality: 4, The detects are from notorious sites. I am seeing more of this type of traffic which is not a good sign. The attackers may be preparing for some new type of attack.

System Countermeasures: 2, Our modern operating systems are hardened and although listening on the internal network, the hosts are among those that are documented as not susceptible to a buffer overflow attack (CERT CA-2000-22).

Network Countermeasures: 4, The firewall blocks access to our host printer daemons from the outside.

Defensive Recommendation

Review the Internet for documented port 515 security alerts. Contact SANS and Snort Users Lists to see if others are encountering similar scans and can provide further insight.

Review the firewall rules to ensure access to port 515 is blocked. Assess exposure.

Discuss detect with Systems Administrators and send a similar packet and observe response.

Multiple choice test question

```
[**] OVERFLOW - Possible attempt at MS Print Services [**]
02/24-09:13:10.871674 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x4A
200.206.162.176:4771-> 142.xxx.xxx.xxx:515 TCP TTL:49 TOS:0x0 ID:20728 IpLen:20 DgmLen:60 DF
******S* Seq: 0xEA16CEF5 Ack: 0x0 Win: 0x7D78 TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 14090605 0 NOP WS: 0
```

In the Snort alert listed above, which of the following is not true:

a) The TCP options include timestamp, windows scale factor, selective acknowledgement and minimum segment size.
b) The minimum segment size of 1460 is consistent with an ethernet network.

c) The packet is a connection attempt to port 515.
d) The UDP packet has an acknowledgement number of 0.

Answer: d)

Snort Detect 5 - Smurf Scanner

```
[gochd@TRONS1 210.196.249.137]$ cat ICMP_ECHO | more
[**] BETA - PING-Broadscan Smurf Scanner [**]
02/23-19:35:24.275296 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
210.196.249.137 -> 142.xxx.xxx.0 ICMP TTL:239 TOS:0x0 ID:12941 IpLen:20 DgmLen:32
Type:8  Code:0  ID:0   Seq:0  ECHO
8E 09 03 00                                     ....

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

[**] BETA - PING-Broadscan Smurf Scanner [**]
02/23-19:35:24.294945 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
210.196.249.137 -> 142.xxx.xxx.8 ICMP TTL:239 TOS:0x0 ID:12941 IpLen:20 DgmLen:32
Type:8  Code:0  ID:0   Seq:0  ECHO
8E 09 03 08                                     ....

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

[**] BETA - PING-Broadscan Smurf Scanner [**]
02/23-19:35:24.317888 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
210.196.249.137 -> 142.xxx.xxx.63 ICMP TTL:239 TOS:0x0 ID:12941 IpLen:20 DgmLen:32
Type:8  Code:0  ID:0   Seq:0  ECHO
8E 09 03 3F                                     ...?

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

[**] BETA - PING-Broadscan Smurf Scanner [**]
02/23-19:35:24.334271 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
210.196.249.137 -> 142.xxx.xxx.64 ICMP TTL:239 TOS:0x0 ID:12941 IpLen:20 DgmLen:32
Type:8  Code:0  ID:0   Seq:0  ECHO
8E 09 03 40                                     ...@

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

[**] BETA - PING-Broadscan Smurf Scanner [**]
02/23-19:35:24.354336 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
210.196.249.137 -> 142.xxx.xxx.127 ICMP TTL:239 TOS:0x0 ID:12941 IpLen:20 DgmLen:32
Type:8  Code:0  ID:0   Seq:0  ECHO
8E 09 03 7F                                     ....

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

[**] BETA - PING-Broadscan Smurf Scanner [**]
02/23-19:35:24.376876 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
210.196.249.137 -> 142.xxx.xxx.128 ICMP TTL:239 TOS:0x0 ID:12941 IpLen:20 DgmLen:32
Type:8  Code:0  ID:0   Seq:0  ECHO
8E 09 03 80                                     ....
```

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

[**] BETA - PING-Broadscan Smurf Scanner [**]
02/23-19:35:24.398169 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
210.196.249.137 -> 142.xxx.xxx.191 ICMP TTL:239 TOS:0x0 ID:12941 IpLen:20 DgmLen:32
Type:8  Code:0  ID:0   Seq:0  ECHO
8E 09 03 BF                                     ....

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

[**] BETA - PING-Broadscan Smurf Scanner [**]
02/23-19:35:24.416193 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
210.196.249.137 -> 142.xxx.xxx.192 ICMP TTL:239 TOS:0x0 ID:12941 IpLen:20 DgmLen:32
Type:8  Code:0  ID:0   Seq:0  ECHO
8E 09 03 C0                                     ....

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

[**] BETA - PING-Broadscan Smurf Scanner [**]
02/23-19:35:24.435866 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
210.196.249.137 -> 142.xxx.xxx.255 ICMP TTL:239 TOS:0x0 ID:12941 IpLen:20 DgmLen:32
Type:8  Code:0  ID:0   Seq:0  ECHO
8E 09 03 FF                                     ....

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

Snort Signature

```
alert icmp any any <> any any (msg:"BETA - PING-Broadscan Smurf Scanner"; itype:
 8; icmp_id: 0; icmp_seq: 0; dsize:4; )
```

Source of trace

The detect is from our company's Internet connection. The sensor is located outside the firewall which filters inbound and outbound Internet traffic.

Detect was Generated by

The Snort Intrusion Detection system 1.7 generated the alert. The alert is reported to the network operator through SnortSnarf v111500.1. The alerts were logged to disk.

Probability the source address was spoofed

1. Reconnaissance Activity

If reconnaissance is the goal, the source IP address is not likely spoofed. This is then a mapping exercise to identify valid broadcast addresses, live hosts and hosts that respond with echo replies. The goal of the reconnaissance may be to include valid broadcast IP addresses in a future Denial of Service attack against a future spoofed source IP. The scanner may also be assuming that most companies receiving the scan will view the source as the victim of the attack rather than a scanner.

The source IP for this scan resolves to the Support System Limited Company of Japan. The host may also be compromised.

2. True Smurf Attack

There is a high probability that the source address was spoofed. The smurf attack mechanism is designed to attack the source IP address of the packet.

ICMP echo requests are sent to various broadcast ports at numerous destination IP addresses. The destination hosts respond with echo replies to the source IP address identified in the packet.

Description of Attack Scenarios

1. Reconnaissance Activity

This is a mapping exercise to identify valid broadcast IP addresses and hosts that respond with echo replies. The reconnaissance could be locating hosts with active broadcast addresses. The goal is to include those hosts that respond in a future DOS attack against a future spoofed source IP. The scanner may be relying on the fact that most companies receiving the scan will view the source as the victim of the attack rather than a scanner.

2. True Smurf Attack

The smurf attack mechanism is designed to attack the spoofed source IP address of the packet.

ICMP echo requests are sent to various broadcast ports at numerous destination IP addresses. The destination hosts respond with echo replies to the source IP address identified in the packet. For a true Denial of Service attack to take place, other networks in addition to our own would have to be involved.

For the attack to achieve some success, the target host must be overwhelmed with responding network traffic to the extent that normal users experience some degradation of service.

Attack Mechanism

Internet protocol (IP) addresses are represented as 32 bit numbers. Number ranges determine if the IP address is a class A, B, or C category of network address. A typical class C address space uses the first 3 octets for the network address and the final octet for the host IP address. The final octet (8 bits) can address up to 256 devices.

A company can further divide the class C address space into 4 more equally sized subnetworks of 64 hosts (4x64=256) each. The first and last IP number in each subnetwork is usually a candidate for a broadcast address (ie. xxx.xxx.xxx.0, xxx.xxx.xxx.63, xxx.xxx.xxx.64, xxx.xxx.xxx.127, etc.). Routers use an IP address as a broadcast address if they forward incoming traffic destined for this address to all hosts on the subnetwork.

The smurf scan sends a series of echo requests to these standard broadcast addresses. If the addresses are being used as broadcast addresses all live hosts on the subnetwork will receive the echo request and return echo replies to the source IP identified in the packet. This has an amplification effect as each one request is intended to return up to 64 echo replies. The amplification increases further with the number of independent sites and companies involved. For hosts that do not exist an ICMP host unreachable response is sent.

The sending host initializes the ICMP request sequence number to 0 and increments it with each new echo request sent. The host generates an echo reply packet for each echo request received. The echo reply is sent with the same identifier and sequence number that was contained in the original echo request.

Correlations

I do not recall observing this source IP in the past. The source IP is not in our IP history data. The IP does not appear on the SANS CID Top Ten List of source IP numbers.

Denial of Service attacks were effectively launched against a number of well known sites in 2000.

Evidence of Active Targeting

There is some evidence of active targeting. Our IP address was identified as the destination. A lot of prior reconnaissance does not appear to have been performed because our network is a class B address space. The attack was designed for a subdivided class C network. It is interesting to note that one packet was addressed to xxx.xxx.xxx.8 which indicates a check for an even smaller subnetwork of 8 hosts. We have defined subnetworks with netmasks but our broadcast addresses do not for the most part match the IP broadcast addresses scanned for.

Severity

Severity Measure = (Criticality + Lethality) – (System Countermeasures + Network Countermeasures)
= 3 – 3 = 0

This coefficient indicates that based on this analysis the detect appears to be an acceptable risk.

Criticality: 3: The echo requests were generated for reconnaissance or to include our hosts as part of a larger smurf attack against a spoofed source IP.

Lethality: 0: The requests even if effective were not significant enough to effect the performance of our network. The IP destination addresses specified did not for the most part agree with our internally defined broadcast addresses.

System Countermeasures: 0: Our hosts will respond to echo requests.

Network Countermeasures: 3: Our router will let in echo requests directed towards valid hosts. The router will not permit echo replies out onto the Internet

Defensive Recommendation

No specific change on our defensive policy is recommended as the routers do not return echo replies to the Internet.

We will continue to monitor for alerts and notify SANS of attacks.

Multiple choice test question

```
[**] BETA - PING-Broadscan Smurf Scanner [**]
02/23-19:35:24.435866 0:2:4A:F6:60:0 -> 0:D0:B7:73:17:37 type:0x800 len:0x3C
210.196.249.137 -> 142.xxx.xxx.255 ICMP TTL:239 TOS:0x0 ID:12941 IpLen:20 DgmLen:32
```

```
Type:8  Code:0  ID:0   Seq:0  ECHO
8E 09 03 FF                              ....
```

In the Snort alert listed above, which of the following is true:

a) The packet uses the TCP protocol.
b) The packet uses the UDP protocol.
c) The echo reply request has a TTL value of 239.
d) The destination IP address is 210.196.249.137.
e) Echo requests are encapsulated by a frame header and trailer when traveling across an ethernet network.

Answer: e)

Assignment 2 - State of Intrusion Detection - Author - David Goch

The Challenges of Intrusion Detection versus Virus Detection - White Paper

The views in this paper are the views of the author. Reference material provided information from which views and conclusions were made.

Ah, the signature file! Yes, the backbone of many a software detection system. Intrusion Detection Systems (IDS) seem to be patterned after their cousins the Virus Detection Software Systems. But there are important differences between Intrusion Detection and Virus Detection.

The first challenge - the intangible versus the tangible

Virus Detection Systems (VDS) were developed to scan mail headers, email message content and file content for standard character strings. If a match occurs the file is quarantined cleaned and / or deleted. Seems pretty simple. Why does this seem so straightforward? Well viruses are for the most part self-contained objects. It is usually easy to tell if a self-contained tangible object is bad.

The link from "the detect" to the object requiring a conclusion is less precise in Intrusion Detection. The end purpose of ID is to form a conclusion about activity based on the content of one or more packets. Although it is easy to tell if a packet does not conform to a standard it is more difficult to deduce a reason that a malformed packet exists. Herein lies the challenge.

The second challenge - integrity expectations are not the same

Virus software assumes the data presented for examination is complete. The file that has arrived looks the way the sender intended. So if the content matches the signature then a virus must be present. Fait complete!

Integrity expectations surrounding network packets are not the same. TCP IP was designed with network reliability in mind. There was an understanding that packets can get corrupted on the network. Routers and hosts recognize that packet corruption occurs. This is why protocol checksums are recalculated, sequence numbers are used and Time to Live (TTL) values are reexamined.

As IDS signatures are added to detect more and more unusual packets they will undoubtedly report more and more network corrupted packets. The challenge for intrusion detection is to determine the difference between a naturally occurring corruption versus an intended one. Some tolerance and thought will have to be incorporated in order to ensure that false positives are minimized.

The third challenge - real time versus pseudo real time.

Virus signature files check content that is copied, downloaded, received and executed on desktops and servers. Virus signature files can also be applied through regularly scheduled scan processes. If the size of the signature file doubles or the scan process takes twice as long this week, no problem the other processes will wait (eg. slower logon) or will operate independently of the scan process.

Not so with Intrusion Detection Systems. These operate in real time mode to examine network traffic as it flies by. In contrast to the Virus Detection Systems, the traffic doesn't wait for IDS signature checking. The detection software must be finished checking the previous packet and be ready to receive the next packet otherwise the packet is gone. This time challenge is characterized through the related issue of packet loss.

IDS copes with packet loss by such methods as:

- reducing packet content checking,
- matching to the first alert as opposed to the most applicable alert,
- avoiding tracking state like relationships between packets.

The fourth challenge - Cause and Effect

Historically virus behavior is predictable. When a virus infects a PC it carries out a predetermined action such as replicating, deleting files or displaying a message. The virus normally operates independently of its creator.

By contrast an intruder is usually directly involved in directing the different stages of an intrusion attempt. The reconnaissance is planned, the results analyzed and the exploits chosen and carried out. The steps and approaches vary and change even as the intrusion is carried out. The goal of the intruder can vary from self-satisfaction to theft or compromise.

The challenge is to acknowledge the involvement of the intruder by attempting to achieve a closer link between this cause and effect. Intrusion detection systems must be more flexible in tracing and linking detected events. Analysis and correlation of IDS signatures could be incorporated into Intrusion Detection Systems. Artificial intelligence theory could be applied to report on predictions as to what the attacker will do next.

The Fifth challenge - Architecture

Virus detection programs are standalone products that use signature files. What happens on host A does not effect the Virus detection program on host B. The programs are situated and run on servers and desktop PCs.

Intrusion Detection Systems are more networks focused. They have sensors located at strategic points around the network. Although the programs may execute on different servers the focus is on the network traffic traveling between the internal network and the

Internet. It is important for all sensors to share data because traffic can enter along one path and leave by another. All related data must be shared to filter out false positives from real attacks. The challenge is one of improving the co-ordination and synchronization mechanisms of the different sensors so that related packets can be combined to create an accurate picture of a conversation.

The Sixth Challenge - What's Normal?

Viruses are just that. There are few false positives. There are no business reasons for the virus to be there. Information Systems staff do not use viruses in the normal performance of their jobs.

The science of Intrusion Detection is different in this regard. Remote access, scans, pings, load balancers etc. do occur naturally for business reasons. Technical and Network Support staff use several reconnaissance tools in their daily work. Developers may run nmap to harden development servers. 3DNS programs are run by web servers to assure response time for customers. These activities are reflected in the network traffic that the sensor sees and yes it does generate alerts. This is a challenge for Intrusion Detection. How can the false positives be cleanly separated from the real detects.

The Seventh Challenge - Management Support

The manager can see the effects of a virus on his personal computer. He sees first hand the running of a periodic scan, the resident program icon and of course the radio and television news releases when a new virus occurs.

Managers do not normally see the IDS consoles. The intrusions are isolated on your network. There is not normally the same radio and television coverage and excitement. The everyday user does not feel a part of the event. The challenge for the IDS Analyst is to simplify the subject and raise the visibility of IDS. Management support is needed to ensure sufficient funding is available for the IDS effort.

The Eighth Challenge - What's this Signature for?

Virus signatures are straightforward. There is a direct link to a need. A virus is discovered. The signature is developed. There is never a reason to remove a virus signature, as the virus can potentially always be a threat.

IDS signature development is more of a challenge. Reconnaisance and attack exploits are developed to take advantage of weakness in application software. Vendors continually upgrade and patch their software to remove weaknesses. The challenge for the IDS signatures is to ensure that they are well documented. The reasons for each signature should be clearly stated. This provides the customer with the ability to tailor signature files by removing signatures that don't apply to his environment.

Although Snort provides rules for inspection, commercial IDS software does not always do this. This sometimes makes it difficult to identify why an alert is generating for a seemly valid packet. The challenge for all is to ensure that documentation exists that explains the reasons for signature in enough detail so that as targeted software is upgraded these signatures can be dropped.

The Ninth Challenge - Our Time

Virus Detection Systems are mature. The support analyst has set procedures and the processes are fairly automated.

Compare this to IDS where the support analyst has to sift through network traffic to eliminate false positives, check host logs for additional information and grep or tcpdump data. Intrusion Detection can involve substantially more time.

Hurrah, the last challenge for today - The Infrastructure

Commercial virus detection and warning systems are fairly mature. The time to signature development is fairly short. The Virus Support Analyst has a support structure in place.

Intrusion Detection support processes are new and evolving. The SANS and partners CID database is new this year. It is great to see the efforts and support of pioneers like Steven Northcutt, Judy Kovaks, Marty Roesch, Yves Ottawa and numerous volunteers. It is a challenge for all to mature the IDS infrastructure as well as communicate and share information so that correlation and signature development continues.

References:

1. Network Intrusion Detection - An Analyst's Handbook 2nd Edition - Stephen Northcutt, Judy Novak
2. TCP/IP Illustrated Volume 1 - The Protocols - W. Richard Stevens
3. Track 3 - Intrusion Detection In-Depth Course Material - The SANS Institute - David Hoelzer, Stephen Northcutt, Judy Novak, Martin Roesch
4. Intrusion Detection System - Cisco vendor material - NetRanger
5. Virus Detection System - Network Associate International vendor material

Intrusion Detection Practical Assignment #3

GIAC ENTERPRISES

INTRUSION DETECTION ANALYSIS REPORT

Submitted by : David Goch, March 21, 2001

TABLE OF CONTENTS

Assignment 3 - GIAC Enterprises

Introduction

The company sells fortune cookie sayings over the web. Analyze the intrusion detection data provided.

Approach

- The scan and alert files were downloaded from SANS web site and unzipped. The December data was selected to serve as the representative sample for the analysis.
- Each file was opened to record the date of the data contained therein. The files SnortS11.txt, SnortS13.txt and SnortS14.txt appeared to contain the same day's data (December 21/2000). Only one of the files was included in the analysis.
- The MY.NET. home network was changed to a numeric home network id (cat filec.txt | sed 's/MY.NET./142.9./g' > filecnew.txt).
- The files were ftp'd to server for running into Snortsnarf.
- This achieved limited success as ran out of memory and encountered disk problems. This approach was abandoned. Relied on writing my own perl scripts to process December data.
- The files for December were combined into one file (cat file1 file2 > filec) and downloaded to my Linux laptop.
- Perl scripts were written and applied to the combined scan and alert files to create standard formatted output data files for analysis. The scripts have not been included in the submission but can be forwarded as required.
- Unix sort command was applied to the combined standardized data files to create versions ordered by source IP, destination IP, source port, destination port and alert type.
- Unix grep command was used as required to locate specific records of interest.
- The OOS data files were opened and reviewed through a browser window. A search feature was used to locate specific records of interest. The data in Appendix B was located using this method.

SCANS and ALERTS

SCANS

Scans are normally performed to find out what hosts and services are available and what versions of the software are present. Once reconnaissance is complete the attacker can determine suitable exploits to use. Intrusion detection software, such as Snort, generates alerts as the software compares network traffic against their signature files.

The total number of scans was 702,674. The descriptions were taken from the data files and the categories are as follows:

SYN	259,249
UDP	413,495
SYNFIN	25,835
UNKNOWN	400
INVALIDACK	606
NOACK	752
NULL	481
FULLXMAS	73
FIN	1,503
VECNA	221
XMAS	50
NMAP	9
OTHER	0

Of the 702,674 scans, 441,616 originated from the home network. This suggests a compromised host or hosts on the internal network.

Originated from home network	441,616
Scans from external Class A source IPs	158,516
Scans from external Class B source IPs	32,097
Scans from external Class C source IPs	70,470
Scans from external reserved network IPs	0

Class A IP addresses are from 0.0.0.0 to 127.255.255.255.
Class B IP addresses are from 128.0.0.0 to 191.255.255.255.
Class C IP addresses are from 192.0.0.0 to 223.255.255.255.

The following table lists the top scanners and associated scan types used. Four of the top scanners belong to our host network and may be an indication of potential compromise.

SOURCE IPs for December Scans

IP Address	Number of Scans	SYNFIN	UDP	SYN
142.9.213.186	50,252	-	50,245	7
24.180.134.156	33,502	-	1,601	31,901
142.9.98.200	32,406	-	32,402	4
212.187.94.162	29,530	-	2	29,528
24.4.196.167	29,528	-	-	29,528
142.9.253.24	23,231	-	-	23,231
62.158.93.109	21,920	-	-	21,920
24.29.40.11	18,744	-	-	18,744
142.9.100.230	16,342	-	13,794	2,548
133.1.36.184	15,042	14,941	-	101

Our network hosts sent scans to the following the top 10 destination IP addresses (listed below). From a review (Unix grep command) of the scan data file I concluded that the UDP scans were addressed to a number of different hosts in a portscan for port 28800 from port 28800. In a perfect world with no firewall blocking, a live host that does not listen on the port will respond with an ICMP port unreachable message. The router will respond with a host unreachable message if the host is offline or doesn't exist.

Scans Originating from Internal IP Addresses

DESTINATION IPs receiving internally sourced Scans from HOME NET (assuming no spoofing)

IP Address	Number of Scans	SYNFIN	UDP	SYN
203.164.58.41	6,459	-	6,459	-
216.15.60.112	5,348	-	5,348	-
203.18.238.26	2,125	-	2,125	-
153.39.194.10	1,847	-	1,847	-
165.251.8.125	1,286	-	-	1,286
63.16.12.165	1,279	-	1,279	-
24.22.135.10	1,275	-	1,275	-
209.90.4.89	1,253	-	1,253	-
24.10.165.29	1,242	-	1,242	-
24.24.42.137	1,239	-	1,239	-

The top source ports used by the internally generated scans are shown below. See appendix A for a list of these ports along with some information on what has been known to listen on them. An examination of the local host files is recommended to determine what and if any of these services are present.

Top Source Ports used by Internal Hosts in scanning

Port Number	Total Scans	SYNFIN	UDP	SYN	UNKNOWN	INVALIDACK	NOACK	NULL	VECNA	FIN	FULLXMAS	XMAS	NMAP
28800	57,240	-	57,237	3	-	-	-	-	-	-	-	-	-
9753	32,153	-	32,153	-	-	-	-	-	-	-	-	-	-
6112	32,117	-	32,117	-	-	-	-	-	-	-	-	-	-
0	20,982	1	20,894	1	10	26	31	1	7	3	6	1	1
53	16,581	-	16,581	-	-	-	-	-	-	-	-	-	-
32780	13,790	-	13,786	4	-	-	-	-	-	-	-	-	-
7777	11,233	-	11,233	-	-	-	-	-	-	-	-	-	-
2213	8,720	-	8,718	2	-	-	-	-	-	-	-	-	-
666	8,550	-	8,550	-	-	-	-	-	-	-	-	-	-
9353	7,972	-	7,972	-	-	-	-	-	-	-	-	-	-
7001	7,616	-	7,616	-	-	-	-	-	-	-	-	-	-
137	7,421	-	7,421	-	-	-	-	-	-	-	-	-	-
17771	5,538	-	5,538	-	-	-	-	-	-	-	-	-	-

The top destination ports used by the internally generated scans are shown below. See appendix A for a list of these ports along with some information on what has been known to listen on them.

Top destination ports scanned:

Top Destination Ports used by Internal Hosts in scanning

Port Number	Total Scans	SYNFIN	UDP	SYN	UNKNOWN	INVALIDACK
28800	52,357	-	52,357	-	-	-
27015	46,265	-	46,265	-	-	-
6112	31,397	-	31,137	260	-	-
25	30,650	-	2	30,648	-	-
0	20,894	-	20,894	-	-	-
7778	17,308	-	17,308	-	-	-
53	15,818	-	15,804	14	-	-
2000	11,571	-	63	11,508	-	-
137	7,453	-	7,453	-	-	-
27016	6,877	-	6,877	-	-	-
7000	4,255	-	4,253	1	-	1
17771	3,844	-	3,844	-	-	-

Scans Originating from External IP Addresses

External hosts sent scans to our network to the following the top 10 destination IP addresses (listed below). The scans are categorized as UDP and SYN scans. A SYN packet usually attempts to determine which hosts are listening to TCP/IP. A host the listens will respond with a TCP/IP SYN ACK response if listening and a RESET if not. If the host is not online or does not exist, the router will send an ICMP Host Unreachable message.

Top internal destination IP addresses scanned:

Home net DESTINATION IPs receiving externally sourced Scans (assuming no spoofing)

IP Address	142.9.223.86	142.9.201.78	142.9.202.94	142.9.98.182	142.9.203.98	142.9.98.133	142.9.221.158	142.9.98.174	142.9.98.123	142.9.6.0
SYNFIN	2	1	1	1	2	1	1	1	1	-
UDP	-	1,601	2,619	-	-	5,543	-	2,261	1,774	-
SYN	48,276	24,780	6,666	9,265	7,138	3	4,551	2	1	1,764
UNKNOWN	2	1	1	-	-	-	-	-	-	-
INVALIDACK	2	6	-	-	1	-	-	-	-	-
NOACK	1	3	-	-	2	-	-	-	-	-
NULL	-	7	1	-	-	-	-	-	-	-
VECNA	2	2	-	-	-	-	-	-	-	-
FIN	-	-	1	-	-	-	-	-	-	-
XMAS	-	-	1	-	-	-	-	-	-	-
NMAP	-	1	-	-	-	-	-	-	-	-
Total Scans	48,285	26,402	9,290	9,266	7,143	5,547	4,552	2,264	1,776	1,764

The data was resorted to determine the top ports used to scan with. This provided a clearer picture as to which ports and services are being searched for. The external scanners are sending TCP packets from traditional DNS (port 53) and FTP (port 21) service ports. Perhaps the scanner is trying to establish a session with any listening host that can be located. Zone transfers of address maps take place over TCP using port 53. FTP data is transferred over port 21.

Top Source Ports used by Scanner:

Top Source Ports used by External Hosts in scanning

Port Number	Total Scans	SYNFIN	UDP	SYN	UNKNOWN	INVALIDACK	NOACK	VECNA	FIN
53	17,872	1,259	16,612	-	-	-	-	-	1
21	17,393	17,371	-	-	18	2	1	1	-
109	7,148	7,148	-	-	-	-	-	-	-
50012	1,337	-	1,334	3	-	-	-	-	-
50013	1,285	-	1,285	-	-	-	-	-	-
7777	1,195	-	1,195	-	-	-	-	-	-
7001	881	-	881	-	-	-	-	-	-
2666	878	-	-	878	-	-	-	-	-
38668	847	-	846	1	-	-	-	-	-
38667	754	-	754	-	-	-	-	-	-
138	495	-	495	-	-	-	-	-	-

When we examine the destination ports where attempted access was tried we get a slightly different list. The external scanners are looking primarily for FTP (port 21) services. Once a listening host is found the scanner will attempt an exploit to get access to the FTP service. Port 109 represents an POP2 internet email service.

Top destination ports scanned:

Top Destination Ports used by External Hosts in scanning

Port Number	Total Scans	SYNFIN	UDP	SYN	UNKNOWN	INVALIDACK	NOACK	VECNA	FIN	NULL	XMAS
21	109,379	17,371	2	92,005	-	1	-	-	-	-	-
109	7,165	7,148	4	13	-	-	-	-	-	-	-
5232	2,923	-	1	2,922	-	-	-	-	-	-	-
53	2,173	1,259	34	880	-	-	-	-	-	-	-
113	1,465	-	3	24	-	-	-	-	-	1,438	-
12346	1,104	-	-	1,104	-	-	-	-	-	-	-
7000	794	-	782	12	-	-	-	-	-	-	-
515	690	-	-	690	-	-	-	-	-	-	-
138	511	-	495	16	-	-	-	-	-	-	-
1	393	-	5	384	-	1	-	3	-	-	-
21536	211	-	-	2	22	45	97	-	27	-	18
0	208	2	-	4	11	32	52	92	12	1	2

ALERTS

Intrusion Detection software compares network traffic to signature files and generates alerts for matches. The alerts can represent scans but also exploits. For example, if a well know exploit such as accessing a subseven trojan program on port 27374 occurs, a match to a signature and alert will follow. Alerts are generated to make IDS Analysts, System Administrators and System Owners aware of intrusion attempts.

The course categorizes alert traffic as essentially friendly fire, hostile fire (scan, probe, break-in, denial of service), and false positives. An example of friendly fire could be a technical support analyst that pings a host to determine if it is available. This can generate an alert indicating a potential scan. An example of a false positive would be a web session where the client selected a well known trojan port number in accordance with its normal random selection process for port selection. We can determine that a false alarm exists after examining the data payloads that are sent in the packets.

December ALERTS BY TYPE

Portscans	144,851
Watchlist 00220	29,535
SYN FIN scan	12,453
Printer 515 outside access	4,224
Tiny Fragments	2,976
Sun RPC access	1,839
WinGate attempts	1,255
Watchlist 00222	1,117
Russia Dynamo SANS flash	546
Null scan	491
SNMP public access	244
Sun RPC high port access	171
NMAP TCP Ping	150
Queso fingerprint	149
SMB name wildcard	123
Broadcast Pings	114
SMTP traffic	89
Back Orifice	56
External RPC call	49
Printer 515 inside access	16
NMAP fingerprint	2
site exec - possible wuftp exploit	2
Happy 99 virus	1
Total	200,453

A total of 200,453 alerts were generated from both internal and external traffic. There were 113,564 alerts originating from internal source IP address traffic and 86,889 alerts originating from external source IP address traffic sent to the home network.

The top alerts in terms of number mostly indicate reconnaissance. Any break-in and exploit attempts would be lower in number and would appear near the bottom of the list. The Happy 99 virus and possible wuftp exploit are examples. These are lower in number but higher in severity.

Similar to the scan data I have presented a list of the top hosts (IP addresses) that originated the alerts. These hosts are candidates for more detailed inspection to ensure that they have not been compromised.

SOURCE IPs for December Alerts (both internal and external)

IP Address	Number of Scans	Port scan	Watch 220	SYNFIN	Printer 515
142.9.214.166	24,384	24,384	-	-	-
212.179.79.2	19,784	-	19,784	-	-
142.9.253.24	15,140	15,140	-	-	-
142.9.213.186	7,164	7,164	-	-	-
142.9.100.230	6,177	6,177	-	-	-
147.8.182.157	4,364	268	-	4,096	-
142.9.217.182	4,052	4,052	-	-	-
194.204.224.131	3,326	274	-	3,052	-
142.9.1.3	3,084	3,084	-	-	-
141.211.176.99	2,505	269	-	-	2,236
142.9.97.154	2,477	2,477	-	-	-

Alerts Originating from Internal IP Addresses

The top alerts generated from internal source IP addresses were the result of portscans. As mentioned earlier exploit attemptsare normally lower in number and would not appear on this list. The top home network source IP addresses that generated alerts are listed below:

SOURCE IPs for December Alerts (internal)

IP Address	Number of Scans	portscan
142.9.214.166	24,384	24,384
142.9.253.24	15,140	15,140
142.9.213.186	7,164	7,164
142.9.100.230	6,177	6,177
142.9.217.182	4,052	4,052
142.9.1.3	3,084	3,084
142.9.97.154	2,477	2,477
142.9.1.5	2,472	2,472
142.9.1.4	2,064	2,064
142.9.97.247	1,904	1,904
142.9.97.165	1,824	1,824
142.9.156.110	1,804	1,804

The port scan traverses a number of hosts so the port scan alert only registers the source IP address of the originating scan. This is why the destination is shown as N/A in the table below. For statistical collection purposes the destination address was entered as 0. This is why it is listed as number 1. In the table below other internal hosts are listed as the destination IP addresses of traffic originating from the home net. This is another sign of a possible internal compromise. It is common for an intruder once established to try to compromise neighboring hosts in an attempt to own more systems. The top IP destination address 194.87.6.38 resolves to the Demo Internet Company, Moscow Russia.

Top destination IP with HOME NET source IP addresses (as identified from alert data):

DESTINATION IPs receiving internally sourced Scans from HOME NET (assuming no spoofing)

IP Address	Number of Scans	Port scan	printer 515	Russia	SNMP	SMB
N/A	112,835	112,835	-	-	-	-
194.87.6.38	442	-	-	442	-	-
142.9.101.192	224	-	-	-	197	27
142.9.50.154	39	-	-	-	39	-
216.181.129.185	9	-	9	-	-	-
142.9.14.1	8	-	-	-	8	-
64.23.4.67	3	-	3	-	-	-
24.13.123.8	1	-	1	-	-	-
151.196.73.119	1	-	1	-	-	-
148.243.214.7	1	-	1	-	-	-
131.204.205.101	1	-	1	-	-	-

The port scan reconnaissance technique uses source and destination port of 0. The main goal is to locate live host IP addresses. Refer to Appendix A for a list of services known to be active on these ports.

Top source ports as identified by the alert data:

Top Source Ports used by Internal Hosts

Source Port	Number of Scans	Port scan	printer 515	Russia	SNMP	SMB
0	112,835	112,835	-	-	-	-
6699	442	-	-	442	-	-
137	27	-	-	-	-	27
3575	10	-	-	-	10	-
1025	9	-	9	-	-	-
4114	5	-	-	-	5	-
4721	4	-	-	-	4	-
32777	4	-	-	-	4	-
32776	4	-	-	-	4	-
1179	4	-	-	-	4	-
1138	4	-	-	-	4	-

The port scan reconnaissance technique uses source and destination port of 0. The main goal is to locate live host IP addresses. Refer to Appendix A for a list of services known to be listening on these ports.

Top destination port as identified by the alert data:

Top Destination Ports used by Internal Hosts

Source Port	Number of Scans	Port scan	printer 515	Russia	SNMP	SMB
0	112,835	112,835	-	-	-	-
2478	442	-	-	442	-	-
161	244	-	-	-	244	-
137	27	-	-	-	-	27
515	16	-	16	-	-	-

Alerts Originating from External IP Addresses

A total of 86,889 alerts were detected for externally originated traffic. The source IP addresses were categorized into the following IP subnet class categories.

SOURCE IP Subnet Classes

External Source IP Class A subnets	22,884
External Source IP Class B subnets	11,853
External Source IP Class C subnets	52,152
	86,889

The top source IP addresses for the externally originating network traffic is identified below:

I resolved the top 4 addresses to ISP ISDN Net Ltd. (212.179.79), the University of Hong Kong (147.8.182.157), the Faculte Des Sciences de Casablanca (194.204.224.131) and the University of Michigan (141.211.176.99). Universities and ISP hosts may be compromised, spoofed or serve as a temporary proxy for the hacker. The sites usually have limited resources with which to investigate incidents. They often will not respond to IDS queries.

SOURCE IPs for December Alerts (external)

IP Address	Number of Scans	Port scan	Watch 220	SYNFIN	printer 515	RPC	WinGate	Back orifice	TOTAL
212.179.79.2	19,784	-	19,784	-	-	-	-	-	19,784
147.8.182.157	4,364	268	-	4,096	-	-	-	-	4,364
194.204.224.131	3,326	274	-	3,052	-	-	-	-	3,326
141.211.176.99	2,505	269	-	-	2,236	-	-	-	2,505
212.179.77.20	2,353	-	2,353	-	-	-	-	-	2,353
24.7.86.215.	2,316	2,316	-	-	-	-	-	-	2,316
200.194.102.99	2,056	266	-	1,790	-	-	-	-	2,056
194.197.170.7	1,851	271	-	1,580	-	-	-	-	1,851
212.179.44.105	1,517	-	1,517	-	-	-	-	-	1,517
216.99.200.242	1,377	1,363	-	-	-	10	1	3	1,377
24.191.63.215	1,369	1,362	-	-	-	1	6	-	1,369

The external IP hosts directed their reconnaissance and exploit activities towards the following home net host addresses. The port scan was directed at a number of home net hosts. For statistical record keeping these were all represented as home net host IP address 0 (N/A in the table). The top destination IP host should be reviewed and vulnerability scanned to ensure that they have been properly hardened against potential attacks.

Home net DESTINATION IPs from externally sourced IP alerts

IP Address	Number of Scans	Port scan	watch 220	SYNFIN	tiny frag	RPC	Null
N/A	32,016	32,016	-	-	-	-	-
142.9.225.234	9,312	-	9,309	-	-	3	-
142.9.229.114	5,081	-	5,080	1	-	-	-
142.9.228.214	4,448	-	4,445	3	-	-	-
142.9.1.8	2,329	-	-	-	2,329	-	-
142.9.202.30	2,288	-	2,288	-	-	-	-
142.9.130.187	1,518	-	1,517	1	-	-	-
142.9.98.114	1,226	-	1,221	2	-	-	3
142.9.209.154	859	-	858	1	-	-	-
142.9.213.222	803	-	803	-	-	-	-

The next table lists the top source ports used by the externally generated traffic that was detected. Appendix A lists any services that are known to be associated with these ports.

Top Source Ports used by External Hosts

Source Port	Number of Scans	Port scan	watch 220	SYN FIN	tiny frag	RPC	Null	Broadcast	NMAP
0	35,202	32,016	-	-	2,976	-	96	114	-
38318	9,307	-	9,307	-	-	-	-	-	-
109	7,148	-	-	7,148	-	-	-	-	-
40227	5,078	-	5,078	-	-	-	-	-	-
31835	2,540	-	2,540	-	-	-	-	-	-
21	2,431	-	-	2,430	-	1	-	-	-
31012	1,905	-	1,905	-	-	-	-	-	-
4000	1,788	-	-	-	-	1,788	-	-	-
9055	1,580	-	-	1,580	-	-	-	-	-
1	1,518	-	1,517	-	-	-	1	-	-
53	1,277	-	-	1,259	-	2	-	-	16

The next table lists the top destination ports used by the externally generated traffic that was detected. Appendix A lists any services that are known to be associated with these ports. Destination port 0 is normal for a port scan that is more targeted towards host discovery.

Top Destination Ports used by External Hosts in scanning

Dest Port	Number of Scans	Port scan	watch 220	SYNFIN	tiny frag	printer 515	RPC	Watch 222	WinGate	Russia	Null	Broadcast	NMAP	Queso	wuftp
0	35,215	32,016	-	-	2,976	-	-	-	-	-	105	114	-	4	-
4876	9,525	-	9,525	-	-	-	-	-	-	-	-	-	-	-	-
4967	9,315	-	9,315	-	-	-	-	-	-	-	-	-	-	-	-
109	7,148	-	-	7,148	-	-	-	-	-	-	-	-	-	-	-
515	4,224	-	-	-	-	4,224	-	-	-	-	-	-	-	-	-
6699	3,654	-	3,535	-	-	-	-	-	-	104	11	-	1	3	-
21	2,442	-	-	2,430	-	-	-	10	-	-	-	-	-	-	2
32771	2,010	-	-	-	-	-	2,010	-	-	-	-	-	-	-	-
9055	1,580	-	-	1,580	-	-	-	-	-	-	-	-	-	-	-
2209	1,517	-	1,517	-	-	-	-	-	-	-	-	-	-	-	-
53	1,322	-	-	1,259	-	-	-	-	-	-	-	-	63	-	-
1080	1,256	-	-	-	-	-	-	-	1,255	-	-	-	-	1	-

OOS Data Analysis

Further Analysis was conducted on the data provided in the OOS files. The records contain the alert followed by the any data payload bytes up to the snap length that was specified within the Snort application.

Refer to Appendix A for ports identified as unassigned. Traffic was observed using these port numbers. A search was performed in the OOS data to determine if the nature of the service can be identified. For any payloads that were found, an example was included for illustration in Appendix B.

Host 142.9.217.182 appears to have a trojan program installed that listens and converses with external traffic. The covert channel appears to use the TCP/IP flags, options, sequence numbers and possibly Window size fields.

There also appears to be peculiar HTML traffic directed at one or more home net hosts. The external source port is 18245 and the home net destination port is 21536. The directory name cows could be a reference to a hacker cult. The traffic may represent more than 1 unauthorized web server installed. Again the covert channel uses the TCP/IP flag field as a control field.

APPENDIX A

Default Port Numbers and Where Used

Port	Comments
0	Used in fingerprinting OS
1	Tcpmux - test if SGI Irix service is running. Plus Sockets des Troie Trojan
21	FTP, plus several trojan programs use this port
25	SMTP mail service, plus several trojan programs use this port
53	DNS
109	POP2 internet mail
113	identd / auth - used to identify the owner of a connection, plus invisible identd and kazimas trojan
137	Netbios name service, plus msinit trojan
138	Netbios datagram service (Microsoft), plus chode trojan
161	SNMP
515	Print server - sometimes used as an alternate for syslog messages
666	Doom game port, plus several trojan programs use this as a default
1025	no specific service noted, can be any program
1080	SOCKS, WinGate
1138	no specific service noted, can be any program
1179	no specific service noted, can be any program
2000	Remote control program, remotely anywhere installs a webserver to this port (TCP), plus several trojan programs
2209	no specific service noted, can be any program
2213	Kali service
2478	Secure Site Authentication server SSL / SLL
2666	extensis service
3575	no specific service noted, can be any program
4000	ICQ
4114	no specific service noted, can be any program
4721	no specific service noted, can be any program
4876	unassigned
4967	unassigned
5232	unassigned
6112	Battlenet gaming server port
6699	Napster
7000	Subseven and other trojan programs have been known to use. Also TCP xfont, X windows font server
7001	callbacks to cache managers, plus freak88 and freak2k trojans
7777	Napster , plus god message and tini troj trojan
7778	interwise service
9055	unassigned
9353	unassigned
9753	rasadv service
12346	Netbus trojan (TCP) and others have used as a default port
17771	unassigned
21536	unassigned
27015	Valve's half life gaming port
27016	Valve's half life gaming port
28800	unassigned

APPENDIX A (continued)

Default Port Numbers and Where Used

Port	Comments
31012	unassigned
31835	unassigned
32771	Ghost portmapper
32776	RPC spray
32777	RPC walld
32780	Possible RPC service
38318	unassigned
38667	unassigned
38668	unassigned
40227	unassigned
50012	dynamic and / or private port
50013	dynamic and / or private port

APPENDIX B - Suspicious Traffic Patterns

Host 142.9.217.182

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
12/13-16:25:13.714378 142.9.217.182:53 -> 24.147.115.119:4803
TCP TTL:126 TOS:0x0 ID:47661  DF
*1SFRP** Seq: 0xC503A79   Ack: 0x46F423   Win: 0x5010
TCP Options => EOL EOL
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
12/13-18:14:00.866230 142.9.217.182:2340 -> 200.221.100.201:1121
TCP TTL:126 TOS:0x0 ID:14021  DF
2*SFR**U Seq: 0xBF49CCE  Ack: 0xA     Win: 0x5010
TCP Options => EOL EOL
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
12/13-18:14:59.979124 142.9.217.182:2340 -> 200.221.100.201:1121
TCP TTL:126 TOS:0x0 ID:17920  DF
2*SFR**U Seq: 0xBFF    Ack: 0xD432000A   Win: 0x5018
TCP Options => EOL EOL Opt 216 Opt 216 Opt216
Opt 216 Opt 216 Opt216 Opt 216 Opt 216 Opt216
Opt 216 Opt 216 Opt216 Opt 216 Opt 216 Opt216
Opt 216 Opt 216 Opt216 Opt 216 Opt 216 Opt216
Opt 216 Opt 216 Opt216 Opt 216 Opt 216 Opt216
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
12/15-01:50:56.798558 142.9.217.182:1133 -> 207.172.3.46:119
TCP TTL:126 TOS:0x0 ID:65023  DF
**SFR*A* Seq: 0xAC0003   Ack: 0x13B71B57   Win: 0x5010
TCP Options => Opt 32(32): 2020 2000 0402 F07A
82CD 0014  0000 0000 0000 0000 0000 0000 0000
0000 0000 EOL EOL EOL EOL
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
```

Host 142.9.

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
12/12-07:04:03.233954 61.159.96.161:18245 -> 142.9.253.114:21536
TCP TTL:49 TOS:0x0 ID:35586  DF
2*SFRP*U Seq: 0x2F696D61   Ack: 0x6765732F   Win: 0x6D70
66 20 48 54 50 2F 31 2E 31                          f HTTP/1.1
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
12/15-09:08:39.894842 63.254.34.58:18245 -> 142.9.140.2:21536
TCP TTL:112 TOS:0x0 ID:1537  DF
**SFRP*U Seq: 0x2F436865   Ack: 0x6D333531   Win: 0x6630
5F 69 6E 64 65 78 2E 68 74 6D                         _ index.htm
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
12/19-07:36:06.524229 213.76.8.6:18245 -> 142.9.253.125:21536
TCP TTL:19 TOS:0x0 ID:25857  DF
**SFRP*U Seq: 0x2F7E6473   Ack: 0x63686D69   Win: 0x636F
31 2F 63 6F 77 73 2F 61 73 63 69 69 2E 68 74 6D      1/cows/ascii.htm
6C 20 48 54 54 50                                      1 HTTP
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
12/12-21:01:32.956774 61.147.75.4:18245 -> 142.9.6.7:21536
TCP TTL:112 TOS:0x0 ID:49921  DF
**SFRP*U Seq: 0x2F7E686F   Ack: 0x736D616E   Win: 0x6561
67 69 66 20 48 54 50 2F 31                            gif HTTP/1
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
```