Global Information Assurance Certification Paper

Paralysis in Analysis
IDS and You

GCIA Certification

Practical Assignment

v4.1

Zachary T. Poncheri
Washington, D.C.
July 2004

PAGE INTENTIONALLY LEFT BLANK

Table of Contents

Abstract

This paper is intended to demonstrate my understanding of and application of the Intrusion Detection System analysis process. This paper provides an overview of the local network and details three specific attacks. Through this report, I hope to both demonstrate my proficiency in Intrusion Detection Analysis and to reemphasize the importance of thorough analysis.

Document Conventions*

When you read this practical assignment, you will see that certain words are represented in different fonts and typefaces. The types of words that are represented this way include the following:

`command`
Operating system commands are represented in this font style. This style indicates a command that is entered at a command prompt or shell.

`filename`
Filenames, paths, and directory names are represented in this style.

`computer output`
The results of a command and other computer output are in this style

URL
Web URL's are shown in this style.

Quotation
A citation or quotation from a book or web site is in this style.

** Document Conventions copied from GIAC Practical Template*

Poncheri, Zachary

Executive Summary

This report is intended to provide an in-depth, comprehensive analysis of MYNET.edu University computer security logs from January 16, 2004, through January 24, 2004. The findings on the local network will be correlated with the trends reported across the Internet for a more relational analysis.

The most difficult part of conducting any analysis is presenting the most significant and relevant data out of a mass of extraneous data. False-positives and data-overload are the two most troublesome problems in every analyst's work. It is important to understand how the various sensors work, what the limitations are, what type of data each is providing, and how each piece fits into the big-picture.

Specifically, this analysis covers Snort alert logs, out-of-spec logs (OOS), and scan logs that have been collected from sensors throughout the University's network. Snort alert logs are generated by sensors on the network that monitor and analyze specific patterns in traffic. When a sensor detects patterns in the traffic that could potentially be malicious, an alarm is triggered and the event is recorded in the alert log. OOS logs are generated by network sensors that monitor malformed traffic. If traffic is identified that does meet normal protocol for that particular type of traffic, an alarm is triggered and the event is also logged. Scan logs are generated by traffic that is seen as potential reconnaissance scanning. Often, a would-be-attacker probes the desired target network for potentially vulnerable hosts. If such nefarious activity is detected it is logged in the scan log files.

The body of this report details three events that are most representative of what's been seen on the MYNET.edu network.

The first analysis shows an Australian computer intensively scanning a specific host on the University network. With some research, it was found that the Australian group this computer is registered under scans hosts across the Internet and publishes any vulnerability of those servers on its web site, Sorbs.net. However, this is a potential danger to the University's network, because these scans consume University bandwidth and have the potential of overloading a system and shutting it down. Therefore, it is recommended that all traffic from that network be blocked from entering the University's network.

The second analysis shows one particular Israeli computer scanning large section of the University's network for hosts that may be infected with a Trojan known as SubSeven. This Trojan is particularly nasty, because it can allow an attacker to view exactly what a user is doing and remotely control the computer without the user's knowledge or consent. It appears that four critical and two non-critical servers may have been compromised. Therefore, it is the recommended that the hosts in question are thoroughly reviewed by their administrators and verified that all virus definitions are up-to-date. If any hosts are indeed compromised, they need to be quarantined from the network, cleaned of any infections, and forensically examined for any missing or damaged data.

The third analysis shows a tremendous number of international hosts probing the majority of the hosts on the University's network for anything running the DameWare service. DameWare is intended to be a tool to allow administrators to control a remote computer. It is unclear at this time if the traffic is related to a new worm or whether this is some sort of distributed scan targeted at an exploitable service. Therefore, it is

recommended that all critical servers are patched to prevent attackers from exploiting the DameWare service. It is also recommended that port 6129, which is being probed, be blocked from entering or leaving the University network to prevent the propagation of this potential worm.

Although there appears to be a few incidents of concern, the overall status of the network is fairly healthy. Monitoring and maintaining a university network presents a unique set of challenges, which must be not restrict the learning of students or sharing of ideas yet maintain a functional and safe environment for carrying out day-to-day business. This report should emphasize the continued importance of monitoring the University's network and remaining vigilant in the enforcement of policy to protect that infrastructure, while being agile in supporting the needs of the students.

Detailed Analysis

Log Files

This analysis details the three most significant events recorded in Snort alert logs, scan logs, and OOS logs between January 16, 2004, and January 25, 2004. The following files are from GIAC Logs at http://isc.sans.org/logs/ See Table 1.

Alert Logs	Scan Logs	OOS Logs
alert.040116	scans.040116	oos_report_040112
alert.040117	scans.040117	oos_report_040113
alert.040118	scans.040118	oos_report_040114
alert.040119	scans.040119	oos_report_040115
alert.040120	scans.040120	oos_report_040116
alert.040121	scans.040121	oos_report_040117
alert.040122	scans.040122	oos_report_040118
alert.040123	scans.040123	oos_report_040119
	scans.040124	oos_report_040120

Table 1. Log Files Analyzed

Network Topology

In conducting the analysis, the true source network of the logs was that identified, but will not be mentioned in this report. However, making this connection was very helpful in conducting the analysis, because it allowed for nslookups on the true IPs to determine the primary function of each device based on name. For example, MY.NET.12.6 resolved to "mxinxx.MYNET.edu" which could then be assumed to be an email server. This method worked for identifying DNS servers, FTP servers, Remote Access servers, and others.

Another key discovery was identifying the Network Operations Center's web server, which through visiting with a web browser, yielded a wealth of publicly accessible information on the actual network topology and the current status of each device. InterMapper® is a great tool for any network administrator, but should never be posted on the Internet completely open to the public for obvious security concerns. InterMapper® gives detail down to the building location of each device. Although this information would be handy for an administrator, it would be an even more useful to an attacker planning a targeted assault. This would probably be more significant reconnaissance to an attacker than a full DNS zone transfer, because the administrators have already identified which devices are important enough to monitor. See Figure 1.

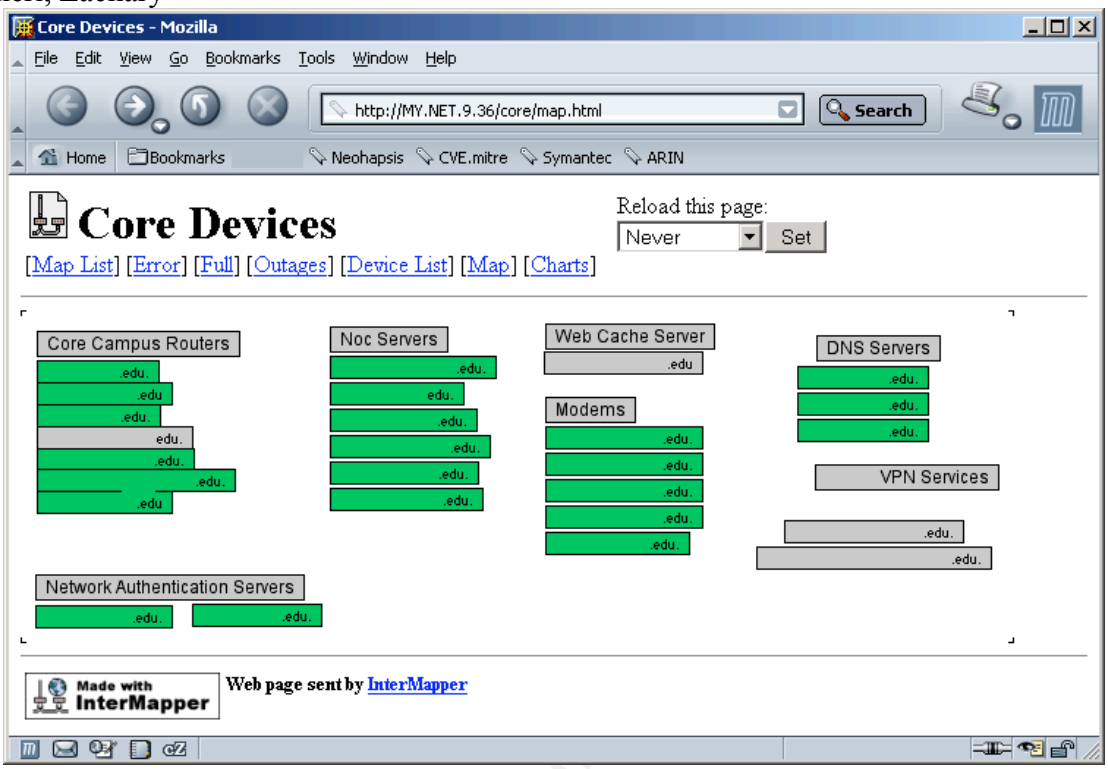


Figure 1. Example of InterMapper® Network Map
(Figure has been obfuscated to protect the source)

Using the methods described above, some of the critical network devices were identified to aid the analysis. See Table 2 for abbreviated list. See Appendix C for complete list. This network is composed of many complex server configurations running a myriad of services. Primarily, the critical servers consist of varying builds of DNS, email, FTP, VPN, and web servers. The network also has many student PCs, which are much harder to control, maintain, and monitor.

Device Name (obfuscated)	**Address**	**Function**
www.MYNET.edu	MY.NET.12.11	Web Server (Primary)
cachexxxx.MYNET.edu	MY.NET.16.42	Web Cache
xxxxcampus.vpn.MYNET.edu	MY.NET.16.106	VPN
xxxxxcampus-gw.MYNET.edu	MY.NET.8.207	Router
mxXXXin.MYNET.edu	MY.NET.25.73	Mail Server
lanxx.MYNET.edu	MY.NET.30.3	Novell Netware Server
Lanxxx.MYNET.edu	MY.NET.30.4	Novell Netware Server
Ftpxxx.MYNET.edu	MY.NET.16.30	FTP Server
MYNET5.MYNET.edu.	MY.NET.1.5	DNS
MYNET3.MYNET.edu.	MY.NET.1.3	DNS
voxx.noc.MYNET.edu.	MY.NET.9.7	Dial-in Server
ciscoxx-dw.MYNET.edu.	MY.NET.2.204	Dial-in Server
xxx.xxx.MYNET.edu.	MY.NET.9.12	Authentication Server
anxxx.MYNET.edu.	MY.NET.30.66	Authentication Server

Table 2. Critical Network Devices

(Table has been obfuscated to protect the source)

Detect 1: SunRPC HighPort Acces followed by TFTP

Alert Logs:

```
01/23-21:53:57.100820 [**] SUNRPC highport access! [**] 203.15.51.59:37021 ->
MY.NET.42.1:32771

01/23-21:54:29.083700 [**] High port 65535 tcp - possible Red Worm - traffic
[**] MY.NET.42.1:65535 -> 203.15.51.59:37021

01/23-21:55:48.102700 [**] SMB Name Wildcard [**] MY.NET.42.1:137 ->
203.15.51.59:137

01/23-21:59:33.067377 [**] TFTP - External TCP connection to internal tftp
server [**] MY.NET.42.1:69 -> 203.15.51.59:37021
```

Scan Logs:

```
Jan 23 22:00:01 203.15.51.59:37022 -> 130.85.42.1:9155 SYN ******S*
Jan 23 22:00:01 203.15.51.59:37022 -> 130.85.42.1:44736 SYN ******S*
Jan 23 22:00:01 203.15.51.59:37022 -> 130.85.42.1:46856 SYN ******S*
[...]
Jan 23 21:59:58 203.15.51.59:37021 -> 130.85.42.1:11346 SYN ******S*
Jan 23 21:59:58 203.15.51.59:37021 -> 130.85.42.1:36373 SYN ******S*
Jan 23 21:59:58 203.15.51.59:37021 -> 130.85.42.1:27542 SYN ******S*
Jan 23 21:59:58 203.15.51.59:37021 -> 130.85.42.1:42459 SYN ******S*
Jan 23 21:59:58 203.15.51.59:37021 -> 130.85.42.1:46410 SYN ******S*
```

OOS Logs:

```
None Related
```

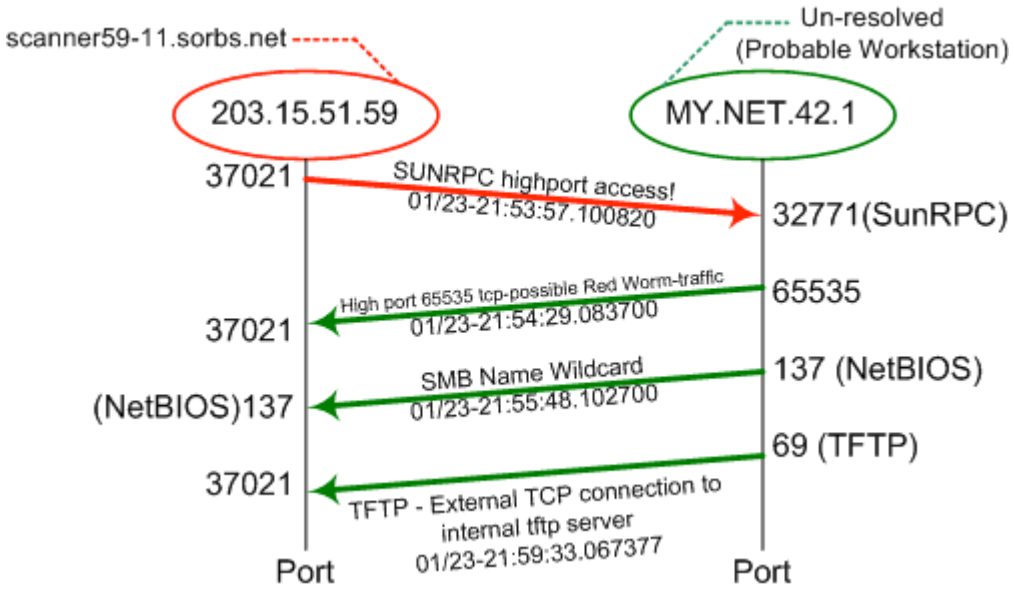


Figure 2. Detect One Link Graph

Why this Detect?

This detect is highly suspicious. Above you can see the attacker, 203.15.51.59, appears

to probe the target for services running via RPC. See Figure 2. Then the targeted machine appears to reply via RPC and SMB with what services it is running along with information about its user accounts. Finally, you can see the extremely suspicious TFTP connection back to the attacker, which could have copied any number of executables or Trojans to the target.

Another reason for focusing on this detect is that traffic is coming from an Australian-based IP. It is improbable that large amounts of traffic originating from an Australian computer has legitimate reason for connecting to a small university in the United States.

Detect Generation

This detect was generated by custom Snort signature alerts on the network monitoring sensors on MY.NET. The exact signatures used to detect this activity were not provided with the alert files. However, it can be assumed from the message fields and port activity that they are similar to the following signatures found on Snort.org and whitehats.com:

RPC portmap listing TCP 32771
(http://www.snort.org/snort-db/sid.html?sid=599)

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 32771 (msg:"RPC portmap
listing TCP 32771"; flow:to_server,established; content:"|00 01 86
A0|"; depth:4; offset:16; content:"|00 00 00 04|"; within:4;
distance:4; content:"|00 00 00 00|"; depth:4; offset:8;
reference:arachnids,429; classtype:rpc-portmap-decode; sid:599;
rev:11;)
```

Custom Snort Signature: HIGH-PORT 65535 TCP
(This signature was written by the author based on content of alert files)

```
alert TCP any 65535 -> any any (msg: " High port 65535 tcp - possible
Red Worm - traffic";)
```

IDS177 “NETBIOS-NAME-QUERY”
(http://whitehats.com/cgi/arachNIDS/Show?_id=ids177&view=signatures)

```
alert UDP $EXTERNAL any -> $INTERNAL 137 (msg:  "IDS177/netbios_netbios-
name-query"; content:   "CKAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA|00 00|";
classtype: info-  attempt; reference: arachnids,177;)
```

Custom Snort Signature: TFTP – External
(This signature was written by the author based on content of alert files)

```
alert TCP $INTERNAL 69 -> any $EXTERNAL (msg: " TFTP - External TCP
connection to internal tftp server ";)
```

Probability of Spoofing

Based on the attack pattern of this detect, there is a low probability of spoofing because the first contact to the target established a TCP connection to send an RPC command and reply, which would not have worked if the IP were spoofed. This can be verified by looking at the initial source port of the attacker, 32701, and the destination port, 32701

of the reply. It can be assumed that a TCP session was successfully established because the reply came back on to the initial source port. Also, the target then initiated a TFTP file transfer back to the address of the attacker further confirming the attacker IP is live. Furthermore, according to whitehats.com, "*the packet that caused this event is normally a part of an established TCP session, indicating that the source IP address has not been spoofed*" (http://www.whitehats.com/info/IDS429).

The attacker IP, 203.15.51.59, was not listed as a well-known proxy on http://www.publicproxyservers.com/. Nslookup did not resolve the IP with this analyst's DNS default servers, but according to dshield.org records, the IP resolves to scanner59-11.sorbs.net (http://www.dshield.org/ipinfo.php?ip=203.15.51.59&Submit=Submit).

Attack Mechanism

The first alert shows a "SUNRPC highport access!" from 203.15.51.59 to the target, MY.NET.42.1, on port 32771, SUNRPC. This is followed by a return from the SUNRPC port on MY.NET.42.1. Notice the source port from the attacker is 37021 in the first alert and in the following alert the targeted host has a destination port of 37021 back to the attacker's IP. This can be seen as evidence that a proper session was established and data was sent back to the attacker. Reference, CVE-1999-0189, bugtraq-id 205, and Xforce-id rpc-32771 (330).

According to Snort.org (http://www.snort.org/snort-db/sid.html?sid=599), a simple command from the attacker could have generated this traffic, which is essentially a query of the target and a return on that query to the attacker. The command, rpcinfo –p IP, is described in Sun Documentation (http://docs.sun.com/app/docs/doc/816-0211/6m6nc675b?a=view) as a probe, which *"shows all RPC services registered with version 2 of the rpcbind protocol on the machine to use."* According to Sun, *"Probe rpcbind on host using version 2 of the rpcbind protocol, and display a list of all registered RPC programs. If host is not specified, it defaults to the local host. … Note that version 2 of the rpcbind protocol was previously known as the portmapper protocol."*

The third alert recorded NetBIOS information sent back to the attacker, which was probably the targeted machine advertising its domain information and user accounts. According to whitehates.com (http://www.whitehats.com/cgi/arachNIDS/Show?_id=ids177&view=research), the *NETBIOS-NAME-QUERY* alert, similar to *SMB NAME WILDCARD*, is triggered by the UNIX samba command "mnblookup –A", which returns the following information:

1. *The NetBIOS name of the server.*
2. *The Windows NT workgroup domain name.*
3. *Login names of users who are logged into the server.*
4. *The name of the administrator account if they are logged into the server.*

The fourth alert appears to show an established TFTP session indicating that the attacker may have successfully triggered a remote TFTP GET back to the attacker, which could have downloaded any number of Trojans. Once a Trojan is installed, the computer is compromised and could be remotely controlled or used as a zombie to attack another target.

Correlation/Evidence of Active Targeting

Although the attacker's IP from this detect could not be correlated with any entries in the OOS logs, it can be correlated between detects found in the alert logs with records

logged in the scans log. There were 21,267 scans on 21,266 unique ports logged, which were all from 203.15.51.59 to MY.NET.51.59 between the times of 21:51:17 and 22:01:42 on January 23, 2004.

The specific *SUNRPC HIGHPORT ACCESS* alert in this detect does not fall at the time of the greatest spike of this signature on the local network, which was January 19t, 2004. See Figure 3.

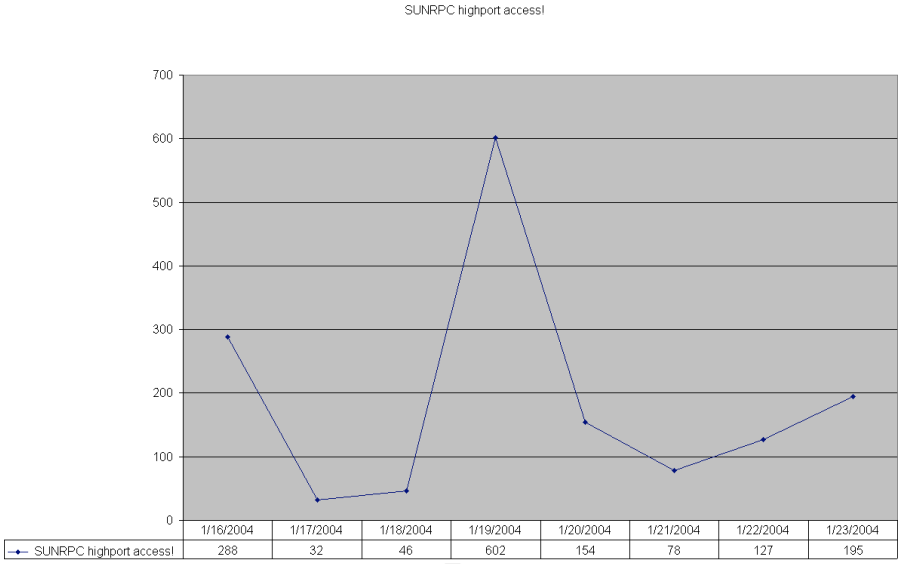


Figure 3. MYNET Alert Log: SUNRPC Highport Access

According to the Internet Storm Center's Port Reports for the timeframe of this analysis, there was a dramatic increase in both the number of targets and total number of records for all activity on port 32771. This correlates nicely with the increase shown in the alert logs and scan logs on the local network. See Figure 4. Note the trend increase starting on January 22 and continuing to increase throughout January 24, 2004

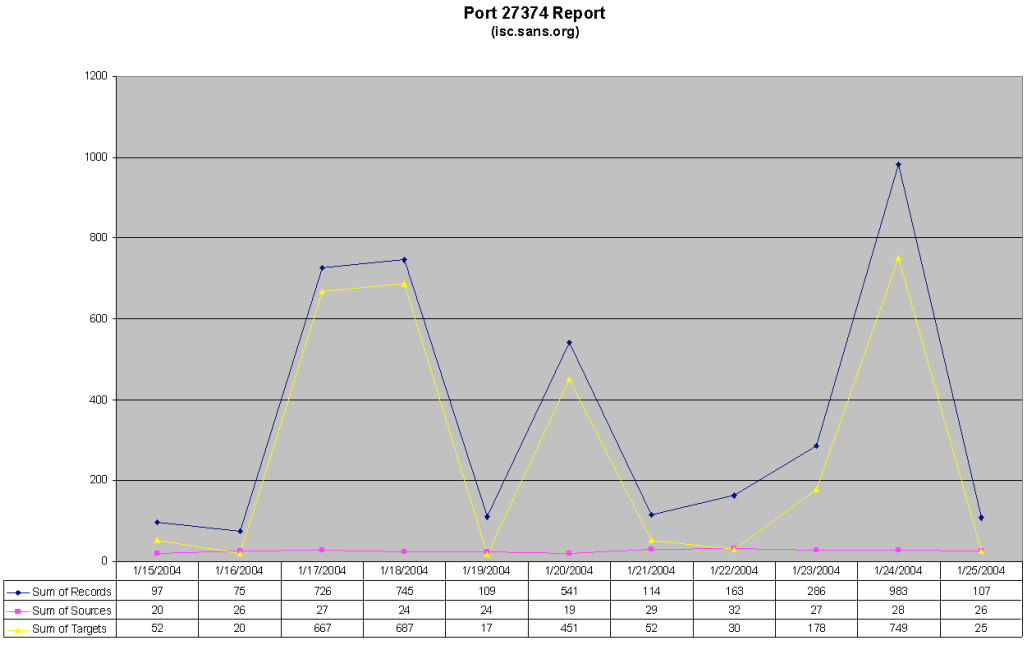


Figure 4. ISC.SANS.ORG Port Report: Port 32771

With further research on the Internet into the source, I found that Spam and Open Relay Blocking System (SORBS.net) is a group of Australian vigilantes determined to end spam by publicly exposing servers that are insecure. According to SORBS.net, their mission is to scan suspected open relays and list them as such on their site. The only way to get a server off the site of vulnerable servers is to pay $50 to either charity or to go towards legal fees to defend the owner of the site in ongoing legal cases.

The philosophy of SORBS.net, as posted in their FAQ (http://www.us.sorbs.net/faq/), states that they feel justified in scanning insecure servers with complete disregard for what damage those scans may cause.

> *8: SORBS makes tests that crashes my server, why...?*
> *It has been shown that certain unpatched configurations of some servers are not stable when some open relay tests are used. SORBS will still use the tests which crash these unpatched server. Running an unpatched server is more of a menace to the Internet as a whole than your individual server crashing. Please ensure your server is patched before you attempt to contact a server using SORBS.*

SORBS.net feels they are protecting the Internet by culling the herd of the weak and whatever they deem unfit for survival. However, the consequences of the SORBS scanning may unwittingly lead to the take down of a hospital server or some other innocent bystander, all in a futile effort to end spam.

Clearly this detect is a case of an actively targeted scan. The source of this detect publicly admits to actively scanning targeted servers for the purpose of advertising those vulnerabilities to the world. In fact, the target of this scan is listed on SORBS.net as have being scanned on January 24, 2004. This explains the over 20,000 scan logs, but does not explain the NETBIOS and TFTP activity.

Although the scan activity may be accounted for, this report can not definitively state the actual intent or exact details of the rather suspicious NETBIOS and TFTP traffic, which followed the port scans by SORBS.net. However, it can be assumed that someone coming from that network has done more than active scanning and possibly initiated a TFTP GET to load something back on to the targeted MY.NET.42.1 node. Keep in mind, the alert, OOS, and scan logs only record that some traffic went across the wire that matched some pattern defined in some signature file. The small gaps of time in the alert files may indicate that some other traffic passed between recorded alerts. This traffic may have then initiated other unrecorded traffic. This analysis is very subjective and can only determine what could have happened or, at best, what appears to have happened.

Severity

Criticality: 1
The primary target of this attack, MY.NET.42.1, appears to be a Solaris workstation with open Samba shares. The nature of the primary function of the target is unknown.

Lethality: 2

Assuming the attacker got the target to download a specific, possibly trojaned file, this attack would be fairly lethal.

System Countermeasures: 0
There are no known system countermeasures implemented on the host to prevent this type of attack.

Network Countermeasures: 1
There doesn't appear to be any network countermeasures such as firewalls or router access control lists to prevent this sort of activity. Only an IDS to monitor and detect this activity is in place, which could serve as post-mortem record of a possible compromised host.

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

Severity = 2

Defensive Recommendation:
The server at MY.NET.42.1 should be configured to disable unauthorized users from accessing rpcbind. Consider implementing TCP wrappers to log connections and limit access control to TCP services. See the CERT.org article, Installing, configuring, and using TCP wrapper to log unauthorized connection attempts on systems running Solaris 2.x (http://www.cert.org/security-improvement/implementations/i041.07.html)

The perimeter firewalls and/or routers should be configured to block rpc-querries and UDP port-137 traffic from an outside network. It would also be advisable to block all traffic coming from SORBS.net, as they are a significant and consistent source of scan traffic, which may result in a denial-of-service attack.

Detect 2: SubSeven Probing and Activity
Alerts Logs:

```
01/23-04:44:16.822701 [**] Possible trojan server activity [**]
217.132.247.46:2594 -> MY.NET.6.15:27374
01/23-04:44:15.625775 [**] Possible trojan server activity [**]
MY.NET.6.15:27374 -> 217.132.247.46:2594
01/23-04:44:16.164443 [**] Possible trojan server activity [**]
217.132.247.46:2594 -> MY.NET.6.15:27374
01/23-04:44:16.164574 [**] Possible trojan server activity [**]
MY.NET.6.15:27374 -> 217.132.247.46:2594
01/23-04:44:16.822701 [**] Possible trojan server activity [**]
217.132.247.46:2594 -> MY.NET.6.15:27374
01/23-04:44:16.822822 [**] Possible trojan server activity [**]
MY.NET.6.15:27374 -> 217.132.247.46:2594

01/23-04:48:41.784396 [**] Possible trojan server activity [**]
217.132.247.46:2051 -> MY.NET.16.90:27374
01/23-04:48:43.745598 [**] Possible trojan server activity [**]
217.132.247.46:2067 -> MY.NET.16.106:27374
01/23-04:48:43.816207 [**] Possible trojan server activity [**]
```

```
217.132.247.46:2075 -> MY.NET.16.114:27374

01/23-06:03:23.605577 [**] Possible trojan server activity [**]
217.132.247.46:3611 -> MY.NET.190.1:27374
01/23-06:03:23.605824 [**] Possible trojan server activity [**]
MY.NET.190.1:27374 -> 217.132.247.46:3611
01/23-06:03:24.220555 [**] Possible trojan server activity [**]
MY.NET.190.1:27374 -> 217.132.247.46:3611
01/23-06:03:24.767746 [**] Possible trojan server activity [**]
MY.NET.190.1:27374 -> 217.132.247.46:3611

01/23-06:03:33.552375 [**] Possible trojan server activity [**]
217.132.247.46:3757 -> MY.NET.190.95:27374
01/23-06:03:33.555007 [**] Possible trojan server activity [**]
MY.NET.190.95:27374 -> 217.132.247.46:3757
01/23-06:03:34.187436 [**] Possible trojan server activity [**]
MY.NET.190.95:27374 -> 217.132.247.46:3757
01/23-06:03:34.831612 [**] Possible trojan server activity [**]
MY.NET.190.95:27374 -> 217.132.247.46:3757

01/23-06:03:33.570851 [**] Possible trojan server activity [**]
217.132.247.46:3759 -> MY.NET.190.97:27374
01/23-06:03:33.573527 [**] Possible trojan server activity [**]
MY.NET.190.97:27374 -> 217.132.247.46:3759
01/23-06:03:34.178108 [**] Possible trojan server activity [**]
MY.NET.190.97:27374 -> 217.132.247.46:3759
01/23-06:03:34.839980 [**] Possible trojan server activity [**]
MY.NET.190.97:27374 -> 217.132.247.46:3759

01/23-06:03:43.834668 [**] Possible trojan server activity [**]
217.132.247.46:3869 -> MY.NET.190.202:27374
01/23-06:03:43.835045 [**] Possible trojan server activity [**]
MY.NET.190.202:27374 -> 217.132.247.46:3869
01/23-06:03:44.347581 [**] Possible trojan server activity [**]
217.132.247.46:3869 -> MY.NET.190.202:27374
01/23-06:03:44.348017 [**] Possible trojan server activity [**]
MY.NET.190.202:27374 -> 217.132.247.46:3869
01/23-06:03:45.000346 [**] Possible trojan server activity [**]
217.132.247.46:3869 -> MY.NET.190.202:27374
01/23-06:03:45.000725 [**] Possible trojan server activity [**]
MY.NET.190.202:27374 -> 217.132.247.46:3869

01/23-06:03:43.843739 [**] Possible trojan server activity [**]
217.132.247.46:3870 -> MY.NET.190.203:27374
01/23-06:03:43.844118 [**] Possible trojan server activity [**]
MY.NET.190.203:27374 -> 217.132.247.46:3870
01/23-06:03:44.457889 [**] Possible trojan server activity [**]
MY.NET.190.203:27374 -> 217.132.247.46:3870
[...]
```

Scan Logs:

```
Jan 23 06:03:24 217.132.247.46:3611 -> 130.85.190.1:27374 SYN ******S*
Jan 23 06:03:23 217.132.247.46:3612 -> 130.85.190.2:27374 SYN ******S*
Jan 23 06:03:23 217.132.247.46:3613 -> 130.85.190.3:27374 SYN ******S*
Jan 23 06:03:23 217.132.247.46:3614 -> 130.85.190.4:27374 SYN ******S*
```

```
[...]
Jan 23 06:03:47 217.132.247.46:3922 -> 130.85.190.253:27374 SYN ******S*
Jan 23 06:03:47 217.132.247.46:3923 -> 130.85.190.254:27374 SYN ******S*
```

OOS Logs:

```
01/19-02:57:12.268979 212.93.133.11:48121 -> MY.NET.6.15:25
TCP TTL:50 TOS:0x0 ID:24115 IpLen:20 DgmLen:60 DF
12****S* Seq: 0x51BC80D5  Ack: 0x0  Win: 0x16D0  TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 15589094 0 NOP WS: 0
```

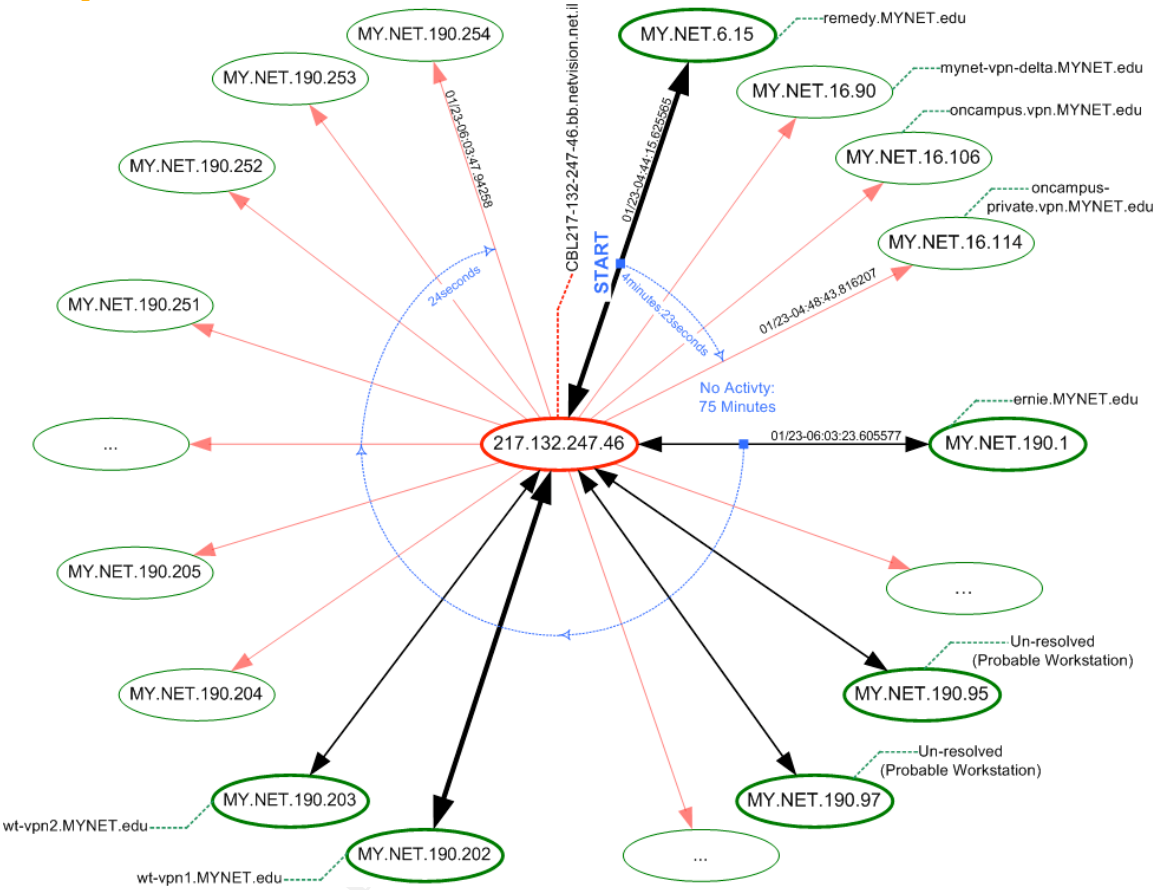


Figure 5. Detect Two Link Graph:
Port 27374 Scan Pattern and Related Activity

Why this Detect?

This detect was chosen for further analysis because the SubSeven Trojan activity detected could be extremely damaging and appears to have a high likelihood of success. See Figure 5. The attacker at 217.132.247.46 scanned 252 distinct addresses on the MY.NET for any device listening on port 27374. Port 27374 is commonly associated with the SubSeven Trojan, which can give an attacker complete remote control over a computer.

Another reason this detect was chosen for further analysis is the source country of this IP. According to the whois records at ripe.net, 217.132.247.46 is registered under an Israeli broadband company, Netvision's Broadband Service. Nslookup

resolves 217.132.247.46 to "CBL217-132-247-46.bb.netvision.net.il". It is improbable that the traffic coming from an Isreali source destined for a university in the US over a port commonly associated with a well-known Trojan is legitimate activity.

Detect Generation

This detect was generated by custom Snort signature alerts on the network monitoring sensors on MY.NET. The exact signature used to detect this activity was not provided with the alert files. However, it can be assumed from the message field and port activity that it is similar to the following signature found on whitehats.com:

> IDS279 "TROJAN-ACTIVE-SUBSEVEN21"
> (http://whitehats.com/cgi/arachNIDS/Show?_id=ids279&view=signatures)
> *alert TCP $EXTERNAL 27374 -> $INTERNAL any (msg: "IDS279/trojan_trojan- active-subseven21"; flags: SA; classtype: system-success; reference: arachnids,279;)*

Probability of Spoofing

It is highly unlikely that the attacker's address was spoofed, because it appears that at least one TCP session was established over port 27374. In order for the attacker to send traffic to a target and receive the response, the attacker would have to use its actual IP. Typically an attacker would spoof their IP if they were trying to launch some sort of denial-of-service attack, because in that case they would not be concerned with receiving any responses.

Attack Mechanism

The alert file generated by campus network sensors show that the attacker specifically targeted port 27374, which is the default port for the SubSeven Trojan, on four addresses in the MY.NET.16 subnet then blindly scanned 248 addresses in the MY.NET.190 subnet on January 23, 2004. It appears the attacker manually probed the first address and then used an automated tool to scan three other addresses, all within 4 minutes and 23 seconds: MY.NET.6.15, MY.NET.16.90, MY.NET.16.106, and MY.NET.16.114. See Figure 5. These addresses are of special concern because they appear to be servers that hold sensitive information and/or provide remote entrance to network resources via VPN. See Table 3.

Device Name (obfuscated)	**Address**	**Function**
remedy.MYNET.edu	**MY.NET.6.15**	**Tech Support Server**
mynet-vpn-delta.MYNET.edu	MY.NET.16.19	VPN
oncampus.vpn.MYNET.edu	MY.NET.16.106	VPN
oncampus-private.vpn.MYNET.edu	MY.NET.16.114	VPN
ernie.MYNET.edu	MY.NET.190.1	Unknown
Un-resolved	MY.NET.190.95	Probable Workstation
Un-resolved	MY.NET.190.97	Probable Workstation
wt-vpn1.MYNET.edu	**MY.NET.190.202**	**VPN**
wt-vpn2.MYNET.edu	MY.NET.190.203	VPN

Table 3. Detect Two Targets of Concern
(Bold text indicates higher volume traffic)

Remedy is a software package used by technical support personnel to manage and process trouble-tickets (http://www.remedy.com/solutions/spm/index.htm). If a server holding customer and network device information such as this were compromised, an attacker would have everything they would need to launch a well-informed, targeted attack or access personal customer records. It appears from the alert files that remedy.MYNET.edu was probed for and replied on port 27374, which may indicate that it was infected with SubSeven and that the attacker may have executed some sort command.

Approximately four minutes after establishing a TCP session with the remedy.MYNET.edu server, the attacker then probed three very specific servers in less than one second. This behavior may indicate that after spending a few minutes reviewing information retrieved from remedy.MYNET.edu, that the attacker may have launched some sort of automated tool to probe three additional servers of interest. However, these three servers did not reply on port 27374 nor did the network sensors record any activity originating from those servers over the scope of this analysis.

The attacker did not trigger any additional signatures until 75 minutes later when he used an automated tool to again attempt to locate devices listening on port 27374. This means the attacker could have been searching for devices that have already been infected with SubSeven. Once an infected device is identified, the attacker could gain unauthorized access to that machine without the owners' knowledge and could have executed any command desired. Although there is only, at most, six alerts triggered per device it would be enough to install or execute any other backdoor that may not have been detected by a signature specifically looking for port 27374 activity.

Of all 252 hosts scanned for port 27374, 6 addresses replied on that port, which indicates a TCP session may have been established. See Figure 5. It can then be assumed that they were also infected with the SubSeven Trojan and that they could now be considered under the control of the attacker.

Correlation/Evidence of Active Targeting

The initial activity of the attacker seemed to be targeted at remedy.MYNET.edu. Next, the attacker targeted three specific VPN servers, which did not reply. That was then followed later by an undirected port 27374 scan for other infected hosts on a separate targeted subnet, MY.NET.190.

The attacker IP, 217.132.247.46, was not found in the OOS logs within the scope of this report, but 248 entries were recorded in the scan log file. There was, however, one OOS log entry which recorded a malformed SMTP packet destined for MY.NET.6.15, the Remedy server. There is also a noticeable increase in the trend of traffic triggering the "*Possible trojan server activity*" signature, which is a direct result of the scan across the MY.NET.190 subnet for port 27374. See Figure 6.

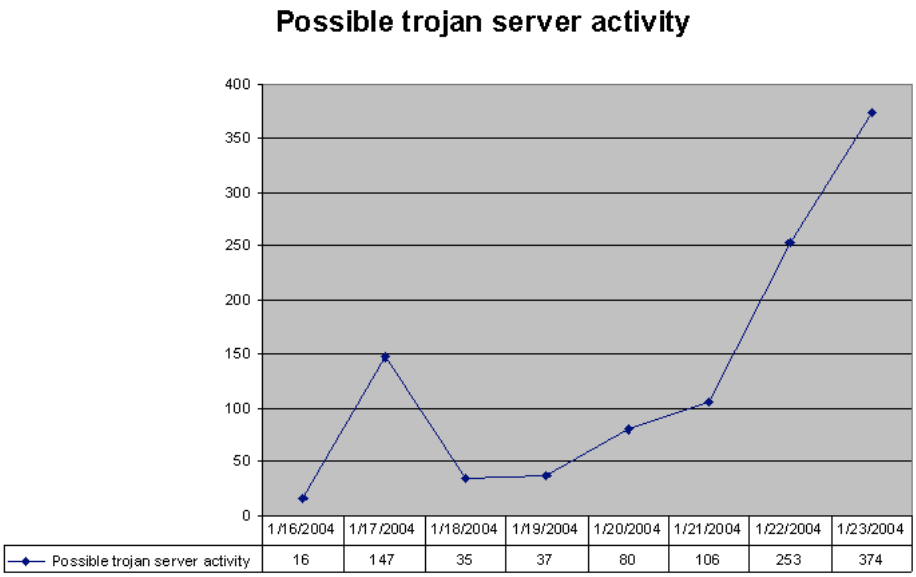


Figure 6. MYNET -- Possible Trojan Server Activity Line Graph

According to the Internet Storm Center's Port Reports for the scope of this analysis, there was a dramatic increase in both the number of targets and total number of reports for all activity on port 27374. This correlates nicely with the increase of alerts recorded on the local network starting on January 20 and continuing to increase throughout January 23, 2004. Compare Figure 6 and Figure 7.

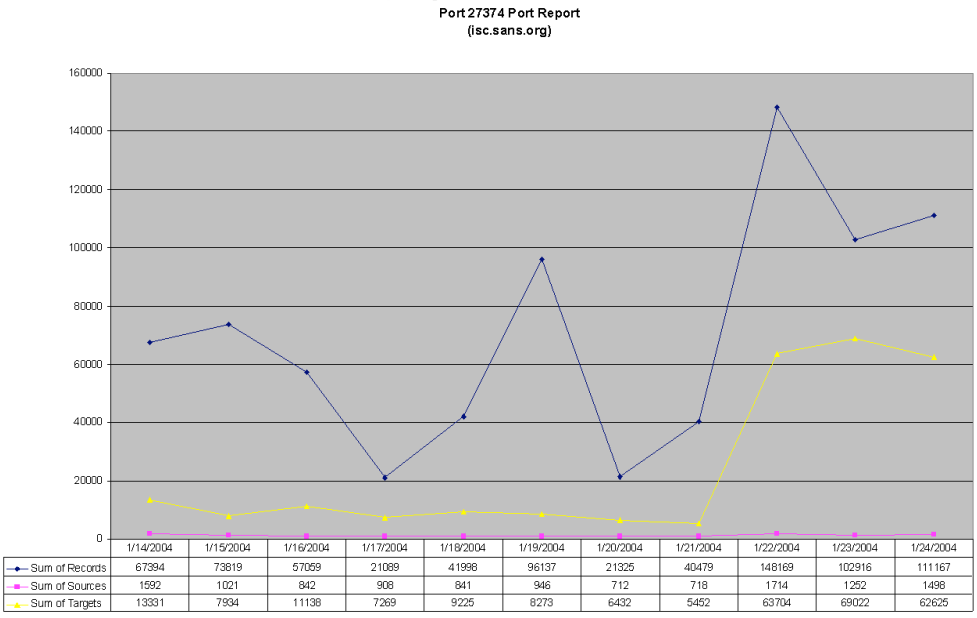


Figure 7. ISC.SANS.ORG Port Report: Port 27374

The attackers' IP was listed as an offender on dshield.org with 8 entries submitted in the past 30 days (http://www.dshield.org/ipdetails.php?ip=217.132.247.46).

See Table 4. Although this may not be the from the same time period or referring to the same port activity, it is still relevant to mention that this same IP has been identified at other sites generating suspicious activity.

Date	Time	Source	Source Port	Target	Target Port	Protocol
2/21/2005	8:42:00	217.132.247.46	1124	not validated	137	17
2/21/2005	8:42:04	217.132.247.46	1124	not validated	137	17
2/21/2005	8:42:19	217.132.247.46	1124	not validated	137	17
2/21/2005	8:42:42	217.132.247.46	1124	not validated	137	17
2/21/2005	8:43:17	217.132.247.46	1124	not validated	137	17
2/20/2005	6:37:42	217.132.247.46	6672	not validated	4672	17
2/20/2005	7:07:39	217.132.247.46	6672	not validated	4672	17
2/20/2005	7:37:24	217.132.247.46	6672	not validated	4672	17

Table 4. Dshield.org Activity: 217.132.247.46

Severity

Criticality: 5
The servers possibly affected by this attack could be considered critical to University operations. One server is critical for maintaining tech support functionality, four servers perform needed VPN functionality, and three servers perform unknown functions.

Lethality: 4
If the SubSeven Trojan were truly installed and operational it would be highly lethal to the hosts compromised, because they would then be under complete control of the attacker.

System Countermeasures: 0
There are no known system countermeasures implemented on the host to prevent this type of attack.

Network Countermeasures: 1
There doesn't appear to be any network countermeasures such as firewalls or router access control lists to prevent this sort of activity. Only an IDS to monitor and detect this activity is in place, which could serve as post-mortem record of a compromised host.

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

Severity = 8

Defensive Recommendation:
All vulnerable systems on the network should have anti-virus software installed, running, and updated. Two of the largest anti-virus software vendors, Symantec and NAI, both have virus definitions to identify and remove various strains of the Subseven Trojan. (http://vil.nai.com/vil/content/v_10566.htm)

Poncheri, Zachary
(http://securityresponse.symantec.com/avcenter/venc/data/backdoor.subseven.html)

Port 27374 traffic could be blocked at the edge router or firewall, which would prevent scans into the network that are looking for nodes listening on the port commonly associated with SubSeven. However, this would not prevent SubSeven from being executed or remotely controlled in all cases. According to NAI, later versions of the SubSeven Trojan can be configured to listen on any obscure port (http://vil.nai.com/vil/content/v_10566.htm), which should reemphasize the importance of proper virus protection.

Detect 3: Suspicious DameWare Scans

Alert Log:

```
01/16-21:56:10.756848 [**] MY.NET.30.4 activity [**] 194.185.90.248:3621 ->
MY.NET.30.4:6129
01/16-21:56:10.761132 [**] MY.NET.30.3 activity [**] 194.185.90.248:3616 ->
MY.NET.30.3:6129
[...]
01/16-21:58:25.184266 [**] MY.NET.30.4 activity [**] 194.185.90.248:2197 ->
MY.NET7.30.4:6129
01/16-21:58:25.805442 [**] MY.NET.30.3 activity [**] 194.185.90.248:2190 ->
MY.NET.30.3:6129
[...]
01/16-22:06:14.768228 [**] SMB Name Wildcard [**] MY.NET.150.44:137 ->
194.185.90.248:137
01/16-22:06:20.375939 [**] SMB Name Wildcard [**] MY.NET.150.198:137 ->
194.185.90.248:137
01/16-22:06:31.915867 [**] SMB Name Wildcard [**] MY.NET.153.21:137 ->
194.185.90.248:137
[...]
```

Scan Log:

```
Jan 17 04:53:19 213.224.225.47:4541 -> MY.NET.30.1:6129 SYN ******S*
Jan 17 04:53:19 213.224.225.47:4542 -> MY.NET.30.2:6129 SYN ******S*
Jan 17 04:53:19 213.224.225.47:4543 -> MY.NET.30.3:6129 SYN ******S*
Jan 17 04:53:19 213.224.225.47:4544 -> MY.NET.30.4:6129 SYN ******S*
Jan 17 04:53:19 213.224.225.47:4545 -> MY.NET.30.5:6129 SYN ******S*
[...]
Jan 17 04:58:42 213.224.225.47:3068 -> 130.85.153.214:6129 SYN ******S*
Jan 17 04:58:42 213.224.225.47:3069 -> 130.85.153.215:6129 SYN ******S*
Jan 17 04:58:42 213.224.225.47:3070 -> 130.85.153.216:6129 SYN ******S*
Jan 17 04:58:42 213.224.225.47:3072 -> 130.85.153.218:6129 SYN ******S*
Jan 17 05:00:21 213.224.225.47:1784 -> 130.85.191.253:6129 SYN ******S*
```

OOS Logs:

None Related

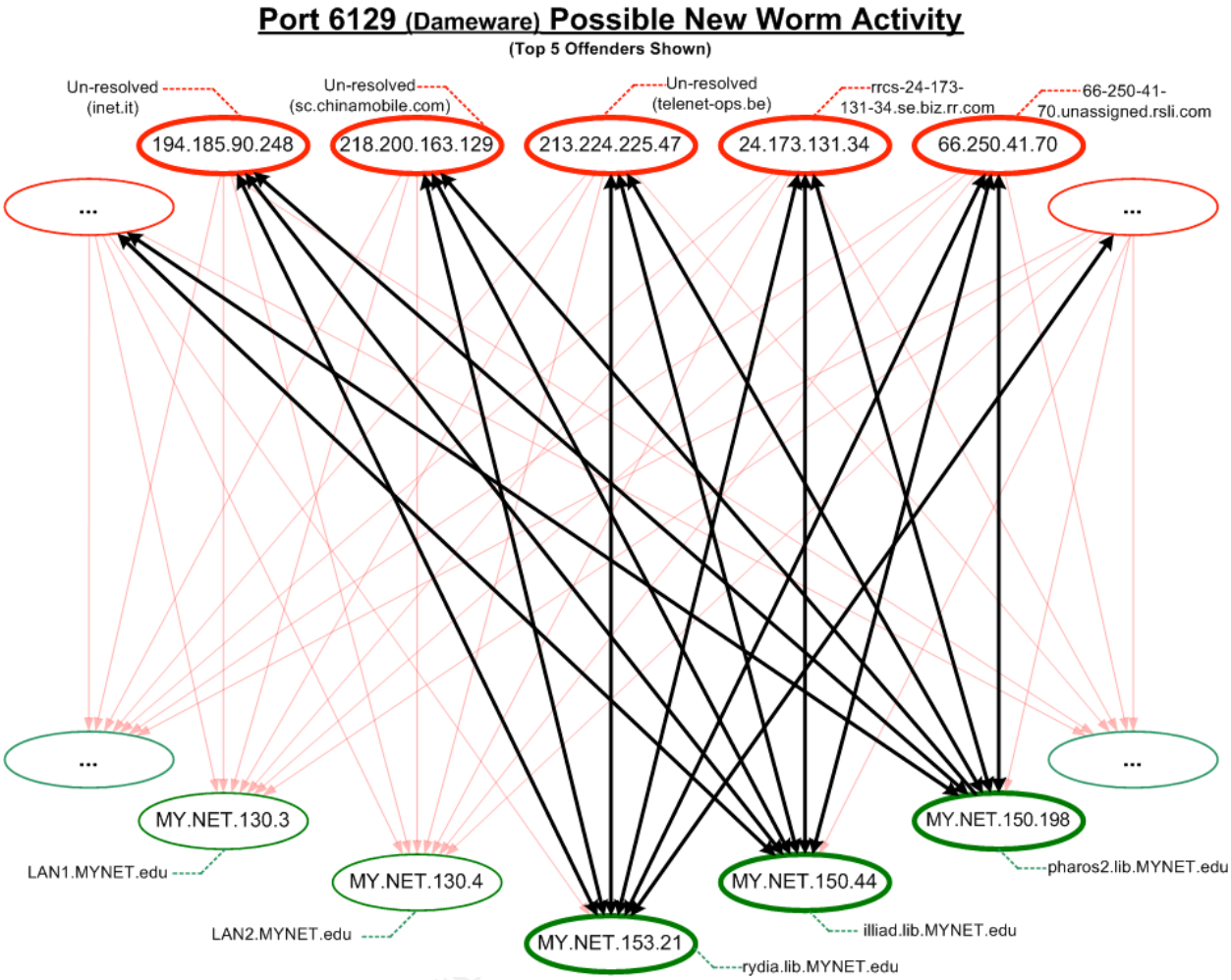


Figure 8. Port 6129 Possible New Worm Activity

Why this Detect?

This detect was chosen for further analysis because it's an interesting case of a possible new worm, which could have easily been overlooked without correlating public reports from isc.sans.org with local traffic.

The only record that this potential worm is slamming the local network is the scan log pre-processor and a few alert entries that were captured by chance. According to the local scan log, 3,351,301 scans on port 6129 were recorded coming from 43,220 unique sources. The top five offenders from the scan log records are shown above in Figure 8. Note the remaining 43,215 source addresses are represented on Figure 8 by the red ovals containing the ellipses. Also, the three hosts shown in bold green ovals in Figure 8 depict three hosts that appear to respond to the port 6129 scans and triggered the *SMB Name Wildcard* signature.

Detect Generation

This detect was generated by custom Snort signature alerts on the network monitoring sensors on MY.NET. The exact signature used to detect this activity was not provided with the alert files. However, it can be assumed from the message field and port activity

recorded in the alert file that the actual signature file used on MYNET.edu is similar to the following signature found on whitehats.com:

IDS177 “NETBIOS-NAME-QUERY”
(http://whitehats.com/cgi/arachNIDS/Show?_id=ids177&view=signatures)
```
alert UDP $EXTERNAL any -> $INTERNAL 137 (msg:  "IDS177/netbios_netbios-
name-query"; content:   "CKAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA|00 00|";
classtype: info-attempt; reference: arachnids,177;)
```

Custom Snort Signature: MY.NET.30.3 Activity
(This signature was written by the author based on content of alert files)
```
alert tcp any any -> MY.NET.30.3 any (msg: "MY.NET.30.3 activity";
\classtype:misc-activity;)
```

Custom Snort Signature: MY.NET.30.4 Activity
(This signature was written by the author based on content of alert files)
```
alert tcp any any -> MY.NET.30.4 any (msg: "MY.NET.30.4 activity";
\classtype:misc-activity;)
```

Probability of Spoofing
It is improbable that the source addresses identified in the alert and scan log files are spoofed, because this detect is associated with possible worm activity and worms general don’t try to hide their source.

Attack Mechanism
At the time of the capture, there were no signatures specifically written to detect the suspected worm, so the only record of the activity is in the scan log and the alert log. See Figure 8. There were 3,351,301 scans for port 6129 recorded coming from 43,220 unique sources between January 16, 2004, and January 24, 2004. See Table 5.

Top 5 Addresses Scanning for Port 6129		
Scan Count	**Source Address**	**Hostname/Network**
28030	66.250.41.70	66-250-41-70.unassigned.rsli.com
28489	24.173.131.34	rrcs-24-173-131-34.se.biz.rr.com
28653	213.224.225.47	Un-resolved (telenet-ops.be)
31977	218.200.163.129	Un-resolved (sc.chinamobile.com)
40455	194.185.90.248	Un-resolved (inet.it)

Table 6. Top 5 Addresses Scanning for DameWare

Reference Figure 8 and note the large number of probes indicated in red unidirectional arrows targeting the DameWare port, TCP 6129, on many servers across the network. Also, note the return traffic indicated in bold black bidirectional arrows from three hosts: pharos2.lib.MYNET.edu, illiad.lib.MYNET.edu, rydia.lib.MYNET.edu. These three hosts could potentially be susceptible to a DameWare vulnerability listed on CVE.mitre.org, CAN-2003-1030, (http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2003-1030). The hosts consistently reply on the NetBIOS port 137, which could be a result of the scan. This activity triggers the *SMB Name Wildcard* signature. There could be

additional traffic between these local hosts being probed and the external hosts, but which were not recorded with the Snort signatures being used at the time.

At first glance, this detect appears to indicate the spreading of a new worm. However, according to the Internet Storm Center Handler Diary dated January 22, 2004 (http://isc.sans.org/diary.php?date=2004-01-22), this traffic appears more likely to be just scan traffic, because there isn't an increase in the number of source addresses being reported. Typically, a worm would show growth in the number of unique source addresses as more hosts become infected. However, other sources can attribute this type of traffic to the Agobot family of worms (http://www.stanford.edu/services/securecomputing/alerts/windows-phatbot-26mar2004.html). See the Sanford University IT advisory below:

> *The Agobot family of worms share the following properties:*
>
> ** Controlled via Internet Relay Chat (IRC), which means that it's a backdoor that does not leave a network port open (ie. can't scan for infected machines directly)*
> ** May infect a machine using a variety of mechanisms:*
> *o User accounts with admin privileges and weak or non-existent passwords*
> *o Microsoft file sharing enabled to allow access to system folder*
> *o Automatic scanning and infection of machines without the RPC/DCOM (MS03-026) or RPC/Locator (MS03-001) Windows operating system patches*
> *o Remotely triggered scanning and infection of machines running Internet Information Services (MS IIS), without the patch for the WebDAV vulnerability (MS03-007).*
> *o Remote detection and communication through the backdoors left by the MyDoom virus (TCP/3127), the Bagel virus (TCP/2745) and the **DameWare exploit (TCP/6129).***
> *o Worm replaces the infected machine's %Windows%\system32\drivers\etc\hosts file with a file that effectively disables access to the Web sites of the major anti-virus vendors (including Symantec, Sophos, McAfee and F-Secure, amongst many others).*

DameWare Mini Remote Control® is a lightweight remote control intended primarily for administrators and help desks for management of desktop systems (http://www.kb.cert.org/vuls/id/909678). According to the US-CERT, DameWare Mini Remote Control prior to version 3.73 is vulnerable to a buffer-overflow that if exploited could give an attacker unauthorized access to execute arbitrary code.

The activity captured in this detect appears to indicate that either a worm is attempting to spread or a large number of zombie hosts are launching many similar scripted scans against a large number of targets. In either case, it appears that the activities recorded are attempts to identify those hosts vulnerable to a specific DameWare exploit and at least three hosts may have been successfully exploited.

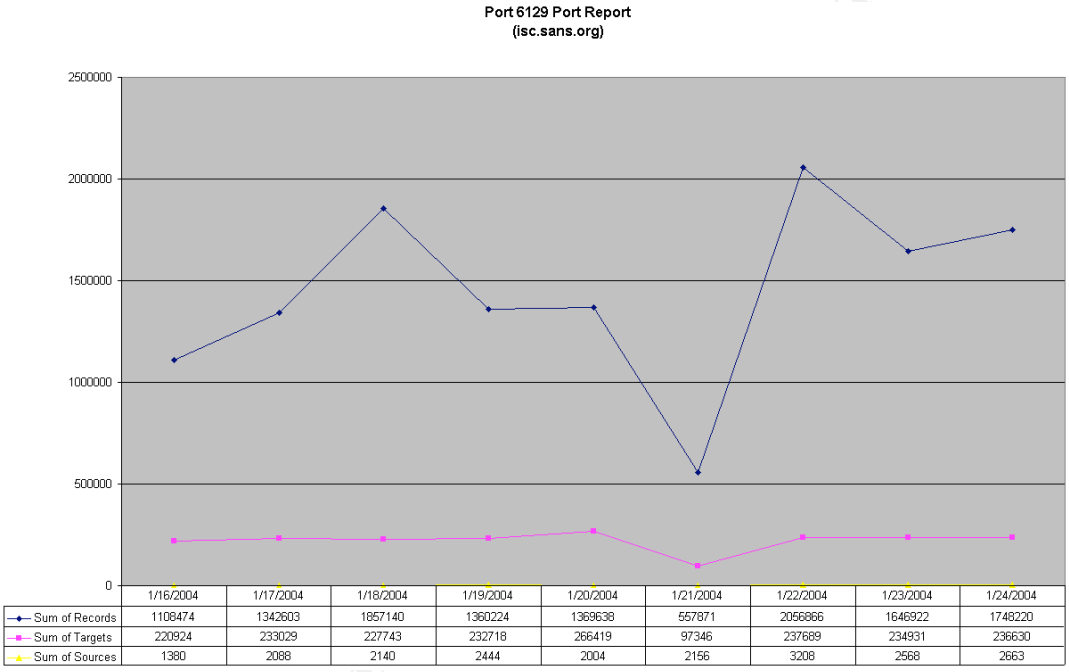
Correlation/Evidence of Active Targeting

There doesn't appear to be evidence of active targeting. According to scan log records, 43,220 external hosts are scanning almost everything in the University address space for hosts vulnerable to the DameWare exploit. The University has a class-C address space, which has about 65,000 addressable hosts, so the scan activity covered approximately 66 percent of the network within 9 days. There were no records in the OOS logs of any hosts involved in the scan or exploit.

The suspicious activity on port 6129 can be correlated with what was reported in the community at the time of the detect. Beginning, January 20, 2004, the Internet Storm Center Handler's Diary noted an increase in port 6129 traffic and requested users

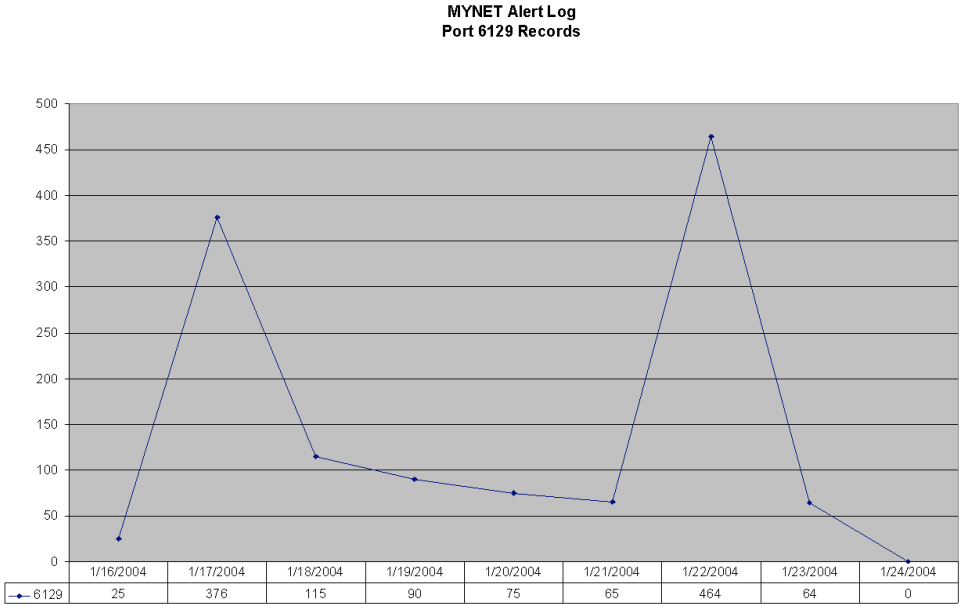
send captures of the traffic. Reporting on port 6129 activity continued throughout January, 23, 2004, when it was determined that the traffic was not due to a worm, but instead just a high volume of persistent scans.

Compare Figure 9 and Figure 10. Figure 9 shows the number of Reports, Targets, and Sources as reported to isc.sans.org. Note the dramatic increase in the number of reports between January, 21, 2004, and January 22, 2004. This increase can be correlated with a similar increase in alert log records on the local network. See Figure 10. There is also a large jump between January, 21, 2004, and January 22, 2004.



	1/16/2004	1/17/2004	1/18/2004	1/19/2004	1/20/2004	1/21/2004	1/22/2004	1/23/2004	1/24/2004
Sum of Records	1108474	1342603	1857140	1360224	1369638	557871	2056866	1646922	1748220
Sum of Targets	220924	233029	227743	232718	266419	97346	237689	234931	236630
Sum of Sources	1380	2088	2140	2444	2004	2156	3208	2568	2663

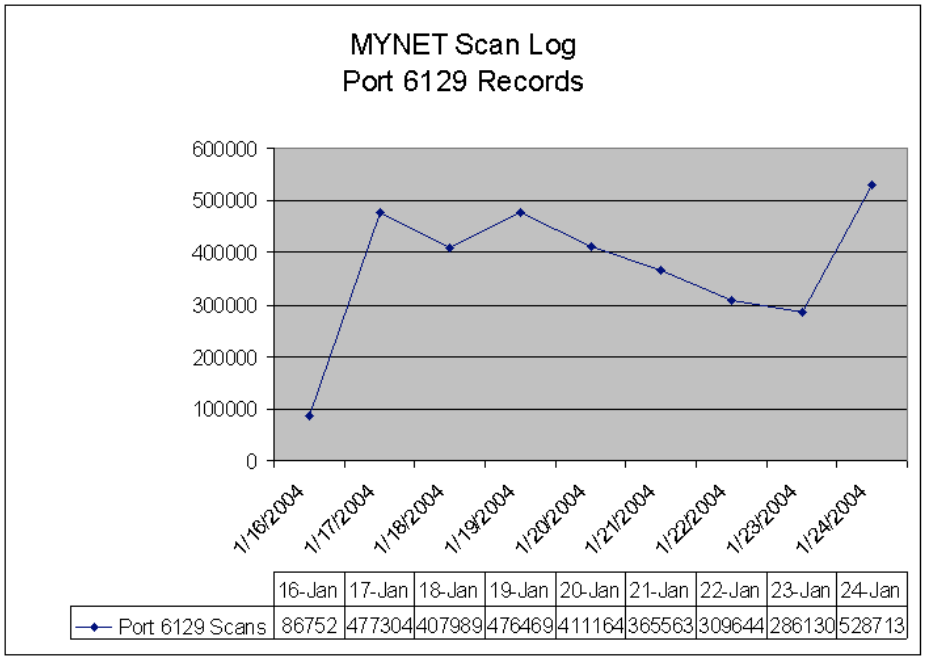
Figure 9. ISC.SANS.ORG Port Report: Port 6129



	1/16/2004	1/17/2004	1/18/2004	1/19/2004	1/20/2004	1/21/2004	1/22/2004	1/23/2004	1/24/2004
6129	25	376	115	90	75	65	464	64	0

Figure 10. MYNET Port Report: Port 6129 Alert Logs

Although Figure 9 and Figure 10 show a direct correlation between isc.sans.org report statistics and MYNET alert statistics, Figure 10 shows an increase in activity slightly later. See Figure 11. Note the increase in MYNET scan log records occurs one day later, between January 23, 2004, and January 24, 2004.



	16-Jan	17-Jan	18-Jan	19-Jan	20-Jan	21-Jan	22-Jan	23-Jan	24-Jan
Port 6129 Scans	86752	477304	407989	476469	411164	365563	309644	286130	528713

Figure 11. MYNET Port Report: Port 6129 Scan Logs

Severity

Criticality: 5
MY.NET.30.3 and MY.NET.30.4 are highly critical Novell Netware servers. This is evident by the fact that the majority of the traffic recorded is destined for the default Novell Netware server port, TCP 524, and the web banner on those servers advertises its Novell services. These hosts were probed on port 6129 but don't appear to be vulnerable, because there are no recorded replies. The hosts, pharos2.lib.MYNET.edu, illiad.lib.MYNET.edu, rydia.lib.MYNET.edu, are all university library servers. According to a google.com site search, the "Illiad" server is necessary for inter-library loans. Presumably the other two lib.MYNET.edu hosts perform various other library functions.

Lethality: 5
If hosts with a DameWare vulnerability were successfully identified and exploited, this attack would be highly lethal and give others remote access to the server.

System Counter Measures: 2
There are IDS signatures specifically monitoring all traffic to and from MY.NET.30.3 and MY.NET.30.4, and the hosts don't appear to be vulnerable to the exploit attempted.

Network Counter Measures: 1
There doesn't appear to be any network countermeasures such as firewalls or router access control lists to prevent this sort of activity. Only an IDS to monitor and detect this activity is in place, which could serve as post-mortem record of a compromised host.

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

Severity = 7

Defensive Recommendation:
All vulnerable systems on the network should have anti-virus software installed, running, and updated. Symantec and NAI, both have virus definitions to identify and remove various strains of related *W32/Polybot.l!irc* (http://vil.nai.com/vil/content/v_101100.htm) and *W32.HLLW.Gaobot.gen* (http://securityresponse.symantec.com/avcenter/venc/data/w32.hllw.gaobot.gen.html) worms.

Port 6129 traffic could be blocked inbound at the edge router or firewall, which would prevent scans into the network that are looking for vulnerable nodes. DameWare is a tool for tech support to remotely administrate a host, which is generally not needed off the local network, so network administrators should consider blocking outbound port 6129 traffic as well.

Network Statistics
See Appendix D for detailed graphs of overall network activity on MYNET in comparison with traffic reported to isc.sans.org.

Top Five Talkers
The statistics for the top five talkers in the alert logs were automatically generated with SnortSnarf. The calculations for the top five talkers in scan and OOS log files were processed with carefully stated grep, awk , and uniq commands. See Appendix A for some scripts used. See Table 6-8 below.

Alert Log Top 5 Talkers		
Count	**Source Address**	**Hostname/Network**
14025	128.171.198.49	s198n49.soc.hawaii.edu
12079	24.35.58.199	cmu-24-35-58-199.mivlmd.cablespeed.com
8647	151.196.123.82	pool-151-196-123-82.balt.east.verizon.net
8604	69.138.237.253	pcp07721328pcs.nrockv01.md.comcast.net
7369	68.163.65.108	pool-68-163-65-108.res.east.verizon.net

Table 6. Alert Log Top 5 Talkers

Scan Log Top 5 Talkers		
Count	**Source Address**	**Hostname/Network**
7370521	MY.NET.1.3	MYNET3.MYNET.EDU
5342233	MY.NET.162.92	oneill-1.MYNET.edu

2760309	MY.NET.111.72	cuereims.MYNET.edu
2749231	MY.NET.84.194	Unresolved (MYNET.edu)
1710914	MY.NET.1.4	MYNET4.MYNET.edu

Table 7. Scan Log Top 5 Talkers

OOS Log Top 5 Talkers		
Count	**Source Address**	**Hostname/Network**
2418	68.54.84.49	pcp0011109240pcs.elkrdg01.md.comcast.net
373	66.225.198.20	Unresolved (scservers.com)
311	81.56.240.245	Unresolved (proxad.net)
158	62.210.155.58	Unresolved (ixo.fr, tiscali.com)
140	207.228.236.26	Unresolved (hopeone.net)

Table 8. OOS Log Top 5 Talkers

Top Five Targeted Ports

The calculations for the top five targeted ports are based on alert, scan, and oos log files processed with carefully stated grep and awk commands, which were exported to comma-separated value files and imported into Microsoft Excel for final tabulation and graphing. See Apendix A for some scripts used.

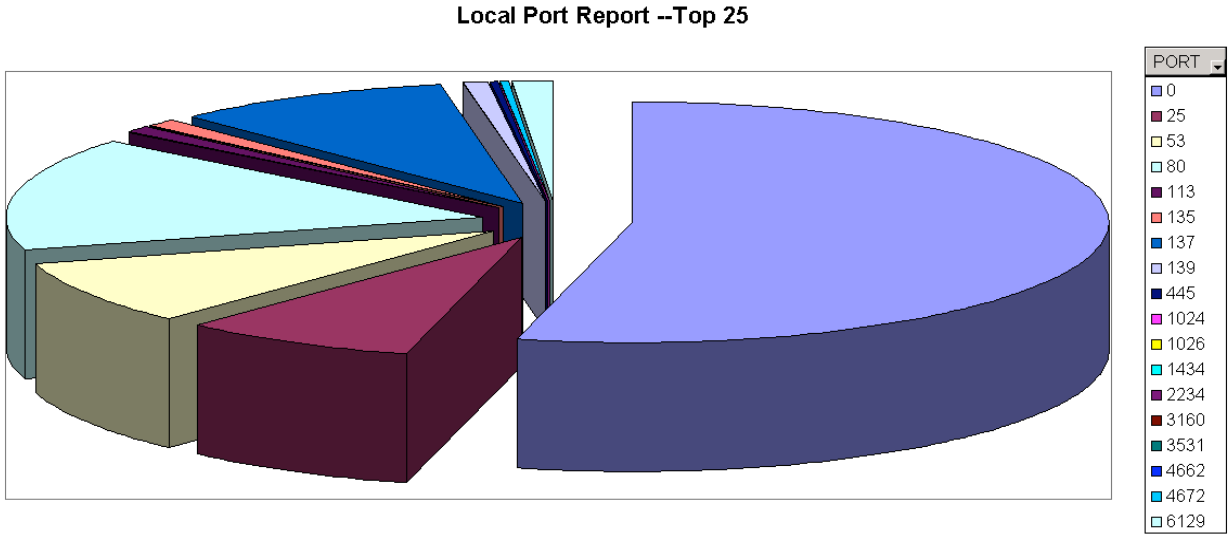


Figure 12. Alert Log Top 25 Destination Ports

Alert Log Top 5 Destination Ports		
Count	**Port**	**Commonly Associated Service**
45227	0	Unknown
12734	80	World Wide Web HTTP
7878	137	NETBIOS Name Service
7206	53	Domain Name Server
6757	25	Simple Mail Transfer

Table 8. Alert Log Top 5 Destination Ports
(http://www.portsdb.org/bin/portsdb.cgi)

Scan Log Top 5 Destination Ports		
Count	**Port**	**Commonly Associated Service**
11175597	135	Microsoft Windows Remote Procedure Call default port
9032706	53	NETBIOS Name Service
3351275	6129	DameWare
1708654	41170	Blubster-- File sharing program for Windows
1602003	25	Simple Mail Transfer

Table 9. Scan Log Top 5 Destination Ports
(http://www.portsdb.org/bin/portsdb.cgi)

OOS Log Top 5 Destination Ports		
Count	**Port**	**Commonly Associated Service**
2540	110	Post Office Protocol - Version 3
1628	25	Simple Mail Transfer
1175	80	World Wide Web HTTP
311	1426	Sais -- Satellite-data Acquisition System 1
179	3672	ChiliASP -- Asp module for Apache servers

Table 10. OOS Log Top 5 Destination Ports
(http://www.portsdb.org/bin/portsdb.cgi)

Top Three Most Suspicious External Sources

The three most suspicious addresses were 66.225.198.20, 81.56.240.245, and 62.225.198.20, because they appeared most often in both the OOS and scan log files. See Table 11 below. Also, these three are way out of the University's geographic area. It is highly unlikely that these sources are conducting any legitimate business. It is more likely that an attacker is trying to compromise hosts on an American university network, which has many vulnerable nodes, lax security, and extremely high bandwidth. Although these addresses don't appear in the alert files to be doing anything directly malicious, they are certainly producing some anomalous activity that shouldn't be overlooked. This traffic is either purely harmless, strange activity or from very skilled attackers that are able to bypass the Snort alert sensors. Never-the-less, traffic originating from these nodes is highly suspicious.

Top 5 in OOS	OOS	Scan	Location	
68.54.84.49	2418	5474	Baltimore, MD	
66.225.198.20	**373**	**701**	**Chicago, IL**	**← Three**
81.56.240.245	**311**	**581**	**Amsterdam**	**← Most**
62.210.155.58	**158**	**343**	**Amsterdam**	**← Suscipicious**
207.228.236.26	140	287	Washington, D.C.	

Table 11. Correlation of Top 5 Entries in OOS and Scan Log Files.

The majority of OOS and scan traffic captured from 66.225.198.20 appears to be destined for the TCP SMTP port on the inbound email server, MY.NET.12.6

(mxin.MYNET.edu). This could be some type of automated email sending script that runs periodically every few minutes.

All OOS and scan traffic captured from the host at 81.56.240.245 appear to be targeting TCP port 1426, which is commonly associated with the Satellite-data Acquisition System protocol (http://www.portsdb.org/bin/portsdb.cgi?portnumber=1426). Also, all OOS and scan traffic captured from the host at 62.210.155.58 appear to be targeting TCP port 80, which is commonly associated with the HTTP protocol (http://www.portsdb.org/bin/portsdb.cgi?portnumber=80).

OOS Logs:

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
01/18-01:49:19.094183 68.54.84.49:38535 -> MY.NET.6.7:110
TCP TTL:51 TOS:0x0 ID:43138 IpLen:20 DgmLen:60 DF
12****S* Seq: 0xEE32F5E5  Ack: 0x0  Win: 0x16D0  TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 181362023 0 NOP WS: 0
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
01/18-01:50:22.847226 68.54.84.49:38536 -> MY.NET.6.7:110
TCP TTL:51 TOS:0x0 ID:7860 IpLen:20 DgmLen:60 DF
12****S* Seq: 0xF1E868FA  Ack: 0x0  Win: 0x16D0  TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 181368398 0 NOP WS: 0
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

The host at 68.54.84.49 has the highest count of OOS and scan log records, but with further analysis it is evident that this is most likely legitimate student activity over a cable modem. The large numbers of OOS logs appear to be erroneously generated from a misconfigured email client. The OOS entries occur about every minute and all to the same port 110 on the same email server, MY.NET.6.7. TCP port 110 is commonly associated with POPv3 (http://www.portsdb.org/bin/portsdb.cgi?portnumber=110). Therefore, it can be concluded that a user at home has their email client configured to check mail once every minute which has some misconfiguration that generates the OOS logs each time.

Analysis Process

I chose to use Cygwin on a Windows XP Pro platform to conduct the bulk of this analysis. This worked well for me, because it gave me the application support and ease of Windows with the power of command-line scripting.

My first step in analyzing this data was to get all the data for the timeframe of interest into one central location in a readable format. To do this, I concatenated all alert, OOS, and scan logs into three distinct collections using the concatenate command, cat, in Cygwin to perform this operation.

```
$ cat alert.040116 alert.040117 alert.040118 alert.040119
alert.040120 alert.040121 alert.040122 alert.040123 >
myscope/alerts.mine
```

The next step was to cut out all the extraneous data from the massive concatenated alert log file. Specifically, “spp_portscan” events were duplicated in both scan logs and alert logs. I used the sed command in Cygwin to delete lines containing

"spp_portscan" from the alert log file.

```
$ sed '/spp_portscan/ d' alerts.mine > alerts.mine.noscan
```

After the alert log file was reduced in size and redundancy, I was able to make some final preparations before running SnortSnarf. I used SnortSnarf Version 021111.1, which I found to be extra fussy with the format and size of the input file. Some lines had to be manually cleaned up with a text editor, but most file preparation work could be done with a few UNIX commands. SnortSnarf does not accept input files with "MY.NET" in the first two octets of the obfuscated IP addresses, which turned out to be a problem with using the alert files provided. I used the sed command to do a global search and replace of the "MY.NET" obfuscation string with "111.111." Although 111.111.x.x is a valid and registered IP range, it did not appear in any of the log files, so it served as a functional placeholder for this analysis.

```
$  sed 's/MY.NET/MY.NET/ g' alerts.mine.noscan > alerts.111
```

Even with a properly formatted input file, I ran into some problems running SnortSnarf. After some troubleshooting, I found that the SnortSnarf is very particular about what version of perl is used. Some of the commands that SnortSnarf is dependent on have been deprecated in the later perl versions. By default, Cygwin comes with perl version 5.8.6, which did not work with SnortSnarf Version 021111.1. To work around the problem, I installed an older windows-based perl Version 5.6.1, which worked fine under Windows command line.

Although I had some initial difficulties getting SnortSnarf to run, it was worth the troubles. Instead of using grep and other commands, a user can just click around to see most info they need. Although the interface is a timesaver with many repetitive tasks, such as whois, DNS, and Dshield lookups, it is limited to how you can view the information. For instance, you can't select an IP and see all instances of that IP in both the source and destination fields under just one view. This would be useful in piecing together traffic to and from a potentially compromised node on the local network. Perhaps this could be incorporated in a future revision. For now, it's easy enough to merge source and destination tables for each IP of interest.

Once the alert data was processed by SnortSnarf, it was time for the actual analysis. I began by categorizing and prioritizing each signature alert. I used Mozilla's html editor to easily modify the index.html file created by SnortSnarf for making notations and visually breaking up the tables. See Appendix B: SnortSnarf Report January 16, 2004, to January 23, 2004. Colors and priorities assigned are subjective and for my own clarification. I've categorized each alert. Then color-coded each alert category based on the lethality of each general type. Lastly, I assigned a priority number for each specific signature alert based on lethality of attack and taking into account criticality of specific servers if a specific server was specified in the signature. For instance, the signature alert "MY.NET.30.3" records all traffic to that server, which leads me to believe it is a highly critical server. Therefore, I assigned a priority of 6, because the lethality of each alert is not apparent but the criticality is obvious.

With the alert signatures categorized and prioritized, I focused my analysis on the highest priority alerts first. Once I identified alerts of interest and singled out a particular

host, I looked at the traffic surrounding the initial alert; both, incoming and outgoing traffic of source and destination within in a few minutes. For example, in Detect One I saw an alert for “SUNRPC highport access!”, so I looked at the activity surrounding that event and noticed some additional suspicious alerts directly follwing.

Throughout this analysis, I developed my own techniques and learned to pay close attention to the relational patterns in traffic. The number of source addresses and number of destination addresses are a good starting point. Presumably, each type of alert would have relational characteristics for each method of attack. I determined the relational pattern of traffic as: MANY-to-MANY, ONE-to-ONE, MANY-to-ONE, or ONE-to-MANY. I also consider the time in relation to the amount of traffic to determine if the traffic is bursty or constant. Bursty traffic is typical of most network activity. However, constant traffic is typical of a large file transfers, VPN connections, ongoing attacks, or other activity.

For instance, large amounts of constant ONE-to-ONE relational traffic could be a targeted DoS attack or a user downloading a large file. Bursty ONE-to-MANY traffic could be indicative of a host on the local network infected with a worm that is attempting to spread or could be legitimate peer-to-peer file sharing traffic, such as bittorrent (http://www.bittorrent.com/introduction.html). On the other hand, constant MANY-to-ONE relational traffic patterns could be a DDoS attack or valid peer-to-peer file sharing activity. Constant MANY-to-MANY relational traffic could indicate a rampant infestation of a new worm across the local and external networks. Learning typical patterns to look for and noting traffic characteristics helped me focus my analysis on more relevant detects.

Along with relational traffic analysis and prioritized alert analysis, I developed some network statistics to attempt a statistical and graphical analysis of alerts over time. To do this, I used command-line tools to parse, sort, count, and format output from alert files into CSV (comma-separated value) files. I then imported the CSV files into Microsoft Excel and used the pivot-table functions to graph the results. Specifically, I used grep, sed, awk, wc, uniq, and sort at the command-line.

Recommendations to Future Students

To those who plan to pursue this certification in the future, here are some of my recommendations to you:

- Start as early as possible
- Learn Perl if you don’t know it already. I struggled through this trying to use bash scripting. Although fine for small jobs, bash is slow and limited in power when processing complex files that are several hundred megabytes.
- Pay close attention to the version of perl that you have installed when trying to run SnortSnarf. I found SnortSnarf Version 021111.1 needs Perl Version 5.6.1.
- Use a PC with dual-heads. Two monitors really help when you are trying to compare logs and research on the Net simultaneously.
- Use a PC with plenty of CPU and RAM. I’ve been working with an old PIII 700MHz, and have spent a lot of time staring at the console waiting for it to free up.

References

American Registry for International Numbers. Whois Database. http://ws.arin.net/cgi-bin/whois.pl?

Asia Pacific Network Information Centre. Whois Database. http://www.apnic.net/apnic-bin/whois.pl

Baker, Andrew R; Caswell, Brian; Poor, Mike. “Snort 2.1 Intrusion Detection” Rockland, MA 2004: Syngress Publishing, 2004.

Bittorrent. http://www.bittorrent.com

Carnegie Mellon University CERT/CC http://ww.cert.org

Dougherty, Dale and Robbins, Arnold. "sed & awk" United States: O'Reilly & Associates, Inc., 1997.

Dshield.org IP Info. http://www.dshield.org/ipinfo.php?=

Google. http://www.google.com

Intermapper Network Monitoring and Alerting Software. http://www.intermapper.com/

Internet Assigned Numbers Authority. Port Number Assignments. http://www.iana.org/assignments/port-numbers

Internet Security Systems, Inc. X-Force Database. xhttp://xforce.iss.net/xforce/search.php

McAfee. Virus Information Library. http://vil.nai.com/vil/default.asp

MITRE Corporation. Common Vulnerabilities and Exposures Database. http://cve.mitre.org/cve/

Neohapsis.com. Archives. http://archives.neohapsis.com/

Novell. Documentation. http://www.novell.com/documentation/a-z.html

Public Proxy Servers. http://www.publicproxyservers.com/.

Portsdb.org The Internet Ports Database. http://www.portsdb.org/

Remedy.com http://www.remedy.com/solutions/spm/index.htm

SANS.org Internet Storm Center. http://isc.sans.org/

SecurityFocus.com Buqtraq Archives. http://www.securityfocus.com/archive/1

SORBS.net. SORBS FAQ. http://www.us.sorbs.net/faq/

Snort.org. Signature Search. http://www.snort.org/cgi-bin/sigs-search.cgi?sid=

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

Sun Microsystems. Bigadmin.
http://www.sun.com/bigadmin/scripts/

Sun Microsystems. Product Documentation: rpcinfo
http://docs.sun.com/app/docs/doc/816-0211/6m6nc675b?a=view

Sun Microsystems. Product Documentation: rpcbind
http://docs.sun.com/app/docs/doc/816-0211

Symantec Security Response. Latest Virus Threats.
http://securityresponse.symantec.com/avcenter/vinfodb.html/6m6nc6759?q=rpcbind&a=view

United States Computer Emergency Readiness Team. http://www.us-cert.gov/

WhiteHats. arachNIDS Database. http://whitehats.com/ids/index.html

Appendix A: Scripts I've Written and/or Adopted

sortIP.sh

Comments:
Pass a filename as the first argument and this script will sort the file by IP and dump the results to a new file with a ".srt" extension.

```
#!/bin/sh
# $1 = file to be sorted
cat $1 | sort -n -t. +0 -1 +1 -2 +2 -3 +3 -4 > $1.srt
```

Reference:
(sort string --http://www.sun.com/bigadmin/scripts/)

Trojan-Search.sh

Comments:
Pass in a port number as the first argument to query a list of known Trojan ports and return the Trojan name.

```
#!/bin/sh
echo "Port:  " $1
cat ./trojan-portlist.csv | grep $1 | awk '{FS="," ; print $1 "
"  $2}'
```

In-grep.sh

Comments:
Use this script to search one file for the contents of another file. First argument should be file containing list of search items. Second argument should be file to be searched. Results of search will be dumped to foundit.txt.

```
#!/bin/sh

# $1 = File containing list of search items
# $2 = File to be searched
 echo "" > foundit.txt
 for i in  `cat $1`
  do
      #echo $i
      #echo $2
      grep $i $2 >> foundit.txt
  done
cat foundit.txt
```

doWhois.sh

Poncheri, Zachary

Comments:
Use this script to pass in a file containing a list of IPs and dump results of all whois queries to a single file, whois_dump.txt

```
#!/bin/sh
# $1 = file to process
OUTFILE="whois_dump.txt"
echo "OUTPUTFILE:  " & $OUTFILE
echo "" > $OUTFILE
for i in  `cat $1`
do
      echo ""  >> $OUTFILE
      echo ""  >> $OUTFILE
      echo "==================="  >> $OUTFILE
      echo $i  >> $OUTFILE
      echo "-------------------"  >> $OUTFILE
      echo ""  >> $OUTFILE
      whois $i >> $OUTFILE

done
```

scan.top5.sh

Comments:
Use this script to process scan log files, calculate the top talkers, and format the output into a clean ASCII report. Pass in the scan log filename as the first argument.

```
#!/bin/sh

#$1 = scan log filename
############### Source ##########################
cat $1 | awk '
BEGIN{ FS = " " }
{
 print ( $4 )
}
END{ }'> scans.src

#Get source addresses
cat scans.src | awk '
BEGIN{ FS=":" }
{
 print ( $1 )
}
END{ }' > scans.src.add

#Get source ports
cat scans.src | awk '
BEGIN{ FS=":" }
{
 print ( $2 )
}
```

```
END{ }' > scans.src.prt


sort --output=scans.src.add.sort.add -n -t. +0 -1 +1 -2 +2 -3 +3
-4 scans.src.add
sort --output=scans.src.add.sort.prt -n scans.src.prt

uniq -c scans.src.add.sort.add | sort -n -r >
scans.src.add.sort.add.cnt
uniq -c scans.src.add.sort.prt | sort -n -r >
scans.src.add.sort.prt.cnt



############### Destination ##########################
cat $1 | awk '
BEGIN{ FS = " " }
{
 print ( $6 )
}
END{ }'> scans.dst

#Get source addresses
cat scans.dst | awk '
BEGIN{ FS=":" }
{
 print ( $1 )
}
END{ }' > scans.dst.add

#Get source ports
cat scans.dst | awk '
BEGIN{ FS=":" }
{
 print ( $2 )
}
END{
}' > scans.dst.prt


sort --output=scans.dst.add.sort.add -n -t. +0 -1 +1 -2 +2 -3 +3
-4 scans.dst.add
sort --output=scans.dst.add.sort.prt -n scans.dst.prt

uniq -c scans.dst.add.sort.add | sort -n -r >
scans.dst.add.sort.add.cnt
uniq -c scans.dst.add.sort.prt | sort -n -r >
scans.dst.add.sort.prt.cnt

############# DISPLAY ############################
```

```
echo "" > Top5
echo "File Proecessed: " $1 >> Top5
echo "" >> Top5
echo "Top 5 SRC Addresses"  >> Top5
echo "====================" >> Top5
echo "  Count   Addr" >> Top5
echo "  -----   ----" >> Top5
head -n 5 scans.src.add.sort.add.cnt >> Top5
echo "" >> Top5
echo "" >> Top5
echo "Top 5 SRC Ports"  >> Top5
echo "====================" >> Top5
echo "  Count   Port" >> Top5
echo "  -----   ----" >> Top5
head -n 5 scans.src.add.sort.prt.cnt >> Top5
echo "" >> Top5
echo "" >> Top5
echo "" >> Top5
echo "Top 5 DST Addresses"  >> Top5
echo "====================" >> Top5
echo "  Count   Addr" >> Top5
echo "  -----   ----" >> Top5
head -n 5 scans.dst.add.sort.add.cnt >> Top5
echo "" >> Top5
echo "" >> Top5
echo "Top 5 DST Ports"  >> Top5
echo "====================" >> Top5
echo "  Count   Port" >> Top5
echo "  -----   ----"  >> Top5
head -n 5 scans.dst.add.sort.prt.cnt >> Top5
```

Appendix B: SnortSnarf Report January 16, 2004, to January 23, 2004

Priority	Category	Signature (click for sig info)	# Alerts	# Sources	# Dests
4	ABNORMAL	External POP to HelpDesk MY.NET.70.50	1	1	1
9	ZOMBIE	[MYNET NIDS IRC Alert] Possible Incoming XDCC Send Request Detected.	1	1	1
10	VIRUS	Bugbear@MM virus in SMTP	1	1	1
10	TROJAN	[MYNET NIDS IRC Alert] Possible trojaned machine detected	1	1	1
8	EXPLOIT	EXPLOIT x86 NOPS	1	1	1
5	ABNORMAL	External POP to HelpDesk MY.NET.70.49	2	1	1
3	RECON	Probable NMAP fingerprint attempt	2	2	2
4	ABNORMAL	External POP to HelpDesk MY.NET.53.29	2	1	1
10	ZOMBIE	[MYNET NIDS IRC Alert] XDCC client detected attempting to IRC	2	2	2
2	FRAG	Fragmentation Overflow Attack	2	1	1
9	ZOMBIE	IRC evil - running XDCC	3	1	2
10	TROJAN	[MYNET NIDS IRC Alert] K\:line'd user detected, possible trojan.	3	1	1
3	ABNORMAL	Traffic from port 53 to port 123	3	1	1
1	SMB/NB	NETBIOS NT NULL session	4	1	2
8	RPC	Attempted Sun RPC high port access	5	3	3

3	ABNORMAL	TFTP - External TCP connection to internal tftp server	5	4	4
10	WORM	NIMDA - Attempt to execute cmd from campus host	8	5	4
2	ABNORMAL	External FTP to HelpDesk MY.NET.53.29	8	6	1
2	ABNORMAL	External FTP to HelpDesk MY.NET.70.49	9	6	1
2	ABNORMAL	External FTP to HelpDesk MY.NET.70.50	11	9	1
9	DOS	DDOS mstream client to handler	11	3	3
3	ABNORMAL	TFTP - External UDP connection to internal tftp server	19	3	3
3	ABNORMAL	TFTP - Internal UDP connection to external tftp server	19	5	5
8	EXPLOIT	EXPLOIT NTPDX buffer overflow	20	8	10
10	TROJAN	RFB - Possible WinVNC - 010708-1	24	11	11
2	ABNORMAL	connect to 515 from inside	25	3	2
8	WORM	[MYNET NIDS] Internal MiMail alert	35	2	33
9	DOS	DDOS shaft client to handler	56	2	2
9	EXPLOIT	EXPLOIT x86 setgid 0	75	55	47
9	EXPLOIT	EXPLOIT x86 setuid 0	84	59	45
8	DOS	FTP DoS ftpd globbing	87	6	2
8	WORM	[MYNET NIDS] External MiMail alert	100	48	1
4	AUDIT	FTP passwd attempt	140	102	1
5	ABNORMAL	TCP SMTP Source Port traffic	166	3	1
6	BOUNCE	ICMP SRC and DST outside	170	64	125

		network			
6	SMB/NB	SMB C access	181	49	3
3	FRAG	Tiny Fragments - Possible Hostile Activity	225	3	4
9	WORM	High port 65535 udp - possible Red Worm - traffic	543	58	50
6	BOUNCE	TCP SRC and DST outside network	587	76	133
7	RPC	External RPC call	924	3	260
10	TROJAN	Possible trojan server activity	1048	75	318
10	TROJAN	[MYNET NIDS IRC Alert] IRC user /kill detected, possible trojan.	1051	50	48
7	RPC	SUNRPC highport access!	1521	43	54
2	RECON	Null scan!	1605	87	120
3	RECON	NMAP TCP ping!	1734	220	325
9	EXPLOIT	EXPLOIT x86 stealth noop	2503	16	15
4	ABNORMAL	connect to 515 from outside	3116	1	1
7	EXPLOIT	EXPLOIT x86 NOOP	3126	502	130
5	ABNORMAL	TFTP - Internal TCP connection to external tftp server	3342	3	3
1	SMB/NB	SMB Name Wildcard	7440	178	658
7	RPC	High port 65535 tcp - possible Red Worm - traffic	18456	1615	10556
2	FRAG	Incomplete Packet Fragments Discarded	19298	101	1092
6	AUDIT	MY.NET.30.3 activity	21618	277	1
6	AUDIT	MY.NET.30.4 activity	78441	544	1

Priority Rating

Poncheri, Zachary

HIGH	10	9	8	7	6	5	4	3	2	1	LOW

Category Color

HIGH											LOW

Colors and priorities assigned are subjective and for my own clarification. I've categorized each alert. Then color-coded each alert category based on the general lethality of each type. Lastly, I assigned a priority number for the lethality of each specific signature alert.

Appendix C: Network Details from University Web Sites

(http://www.gl.MYNET.edu/hardware.shtml)

Hostname	Hardware	OS	Useage	OS Version	Services
Console Service					
console	Intel	Linux	Big scary console server		SSH
console-h1	PowerPC	Linux-embedded	Little less scary console server		SSH
Directory Service					
fett.MYNET.edu	Sun E220R, 2Proc	Solaris	Master Directory Server (ds-master.MYNET.edu)	2.7	SSH, LDAP
dengar.MYNET.edu	Sun NetraT1	Solaris	Slave Directory Server	2.7	SSH, LDAP
ig88.MYNET.edu	Sun NetraT1	Solaris	Slave Directory Server	2.7	SSH, LDAP
Authentication Service					
kerberos2.MYNET.edu	SGI Indy	IRIX	Secondary KDC	6.5	
kerberos.MYNET.edu	SGI Indy	IRIX	Primary KDC	6.5	
Mail Delivery					
mx1del.MYNET.edu	Sun Netra t1	Solaris	Mail Delivery/Lookup	2.8	SSH, SMTP
mx2del.MYNET.edu	Sun Ultra5	Solaris	Mail Delivery/Lookup	2.8	SSH, SMTP
mx3del.MYNET.edu	Sun Ultra5	Solaris	Mail Delivery/Lookup	2.8	SSH, SMTP
mx4del.MYNET.edu	Sun Netra t1	Solaris	Mail Delivery/Lookup	2.8	SSH, SMTP
mx1in.MYNET.edu	Netra X1	Solaris	Mail Delivery/Lookup	2.8	SSH, SMTP
mx2in.MYNET.edu	Netra X1	Solaris	Mail Delivery/Lookup	2.8	SSH, SMTP
mx3in.MYNET.edu	Netra X1	Solaris	Mail Delivery/Lookup	2.8	SSH, SMTP
Outgoing Mail Relays					
mx1out.MYNET.edu	Netra X1	Solaris	Outgoing Mail Relay	2.8	SSH, SMTP
mx2out.MYNET.edu	Netra X1	Solaris	Outgoing Mail Relay	2.8	SSH, SMTP
mx3out.MYNET.edu	Netra X1	Solaris	Outgoing Mail Relay	2.8	SSH, SMTP
IMAP/POP Mail Reading					
mr1.MYNET.edu	Sun Enterprise 250	Solaris	MYNET Remote Mail Access	2.8	SSH, IMAP, POP
mr2.MYNET.edu	Sun Enterprise 250	Solaris	MYNET Remote Mail Access	2.8	SSH, IMAP, POP
mr3.MYNET.edu	Sun Enterprise 250	Solaris	MYNET Remote Mail Access	2.8	SSH, IMAP, POP
mr4.MYNET.edu	Sun Enterprise 220R	Solaris	MYNET Remote Mail Access	2.8	SSH, IMAP, POP
Web Services					
auxwww1.MYNET.edu	SGI O2 Server	IRIX	WebCT CourseWare	6.5.3m	SSH, HTTP(80) on webct.MYNET.edu
auxwww2.MYNET.edu	SGI Origin 200	IRIX	MyMYNET	6.5.10m	SSH, HTTP(80) on my.MYNET.edu, HTTPS on my.MYNET.edu
auxwww3.MYNET.edu	SGI Octane (2x R10k)	IRIX	MyMYNET (your.MYNET.edu)	6.5.16m	SSH, HTTP(80) on your.MYNET.edu, HTTPS on your.MYNET.edu
www4.MYNET.edu	Sun NetraT1	Solaris	virthost.MYNET.edu	2.7	SSH, HTTP(80) on virthost.MYNET.edu
cgi.MYNET.edu	SGI Challenge S	IRIX	cgi.MYNET.edu web area	6.5.4	SSH, HTTP(80)

www5.MYNET.edu	Sun Netra T1	Solaris	web development	7	SSH
www6.MYNET.edu	Sun Netra T1	Solaris	webauth.MYNET.edu	7	SSH, HTTP(80) on webauth.MYNET.edu, HTTPS on webauth.MYNET.edu
www7.MYNET.edu	Sun Netra T1	Solaris	webadmin.MYNET.edu	7	SSH, HTTP(80) on webadmin.MYNET.edu, HTTPS on webadmin.MYNET.edu
www8.MYNET.edu	Sun E250	Solaris	www.MYNET.edu	7	SSH, HTTP(80) on www.MYNET.edu
www9.MYNET.edu	Sun E220R, 1proc	Solaris	userpages.MYNET.edu	7	SSH, HTTP(80) on userpages.MYNET.edu
AFS File/Database Servers					
bfs1.afs.MYNET.edu	Sun E250 1proc	Solaris	AFS File Server (data storage)	8	SSH
bfs2.afs.MYNET.edu	SGI ORIGIN 200	IRIX	AFS File Server (data storage)	6.5.10f	SSH
bfs3.afs.MYNET.edu	SGI ORIGIN 200	IRIX	AFS File Server (data storage)	6.5.10f	SSH
sauvignon.MYNET.edu	SGI Challenge S (1x R4400)	IRIX	AFS File Server (admin / sw installs)	6.5.5f	SSH
smirnoff.MYNET.edu	SGI Origin 200	IRIX	Central and Rem/Ora backups	6.5.16m	SSH
wedge.MYNET.edu	Sun Netra T1 Ac200	Solaris	AFS File Service (software)	2.8	SSH
biggs.MYNET.edu	Sun Netra T1 Ac200	Solaris	AFS File Service (software)	2.8	SSH
hfs1.afs.MYNET.edu	Intel P850	Linux	AFS File Server (User Homes)	7.2	SSH
hfs2.afs.MYNET.edu	Intel P850	Linux	AFS File Server (User Homes)	7.2	SSH
hfs3.afs.MYNET.edu	Intel P850	Linux	AFS File Server (User Homes)	7.2	SSH
hfs4.afs.MYNET.edu	Intel P850	Linux	AFS File Server (User Homes)	7.2	SSH
hfs5.afs.MYNET.edu	Intel P850	Linux	AFS File Server (User Homes)	7.2	SSH
hfs6.afs.MYNET.edu	Netra T1 AC200	Solaris	AFS File Server (User Homes)	2.8	SSH
hfs7.afs.MYNET.edu	Netra T1 AC200	Solaris	AFS File Server (User Homes)	2.8	SSH
db1.afs.MYNET.edu	Sun NetraX1	Solaris	AFS Database Server	8	SSH
db2.afs.MYNET.edu	Sun NetraX1	Solaris	AFS Database Server	8	SSH
db3.afs.MYNET.edu	Sun NetraX1	Solaris	AFS Database Server	8	SSH
Other Stuph					
ds2.gl.MYNET.edu	SGI Challenge S (R4400)	IRIX	NFS File Server	6.5.5m	SSH, NFS, HTTP(80)
news.MYNET.edu	Intel	Linux	Usenet News Service	6.2	SSH, SMTP, NNTP
listproc.MYNET.edu	Netra T1	Solaris	Mailing Lists	2.6	SSH, SMTP, ILP, HTTP
ragnarok.MYNET.edu	SGI Challenge S (R5000)	IRIX	Anon FTP, license service, Proxy Server	6.5.10m	SSH, FTP
jarjar.MYNET.edu	Sun Enterprise	Solaris	Remedy	2.6	SSH

	250				
threepio.MYNET.edu	Sun Enterprise 250	Solaris	Instructional/Academic Oracle Databases	2.6	SSH
hubris.ucs.MYNET.edu	Intel 2xP2	Linux	Development	varies	SSH
curly.MYNET.edu	Sun NetraX1	Solaris	System Logging	8	SSH
alumni.MYNET.edu	SGI O2	IRIX	Alumni Email Accounts	6.5.10m	SSH, TELNET, RLOGIN, SHELL, POP, IMAP, SMTP
MYNET7.MYNET.edu	SGI Origin 200	IRIX	Faculty/Staff UNIX Shell Access, Mail Delivery, Web Service, Remote Mail Access	6.5.16m	SSH, SMTP, TELNET, RLOGIN, SHELL, HTTP(80)
irix2.gl.MYNET.edu	SGI Origin 200 (2x R10000)	IRIX	Unrestricted UNIX Shell Access	6.5.16m	SSH, TELNET, RLOGIN, SHELL
linux1.gl.MYNET.edu	2x P850	Linux	Unrestricted UNIX Shell Access	7.2	SSH, TELNET, RLOGIN, SHELL
linux2.gl.MYNET.edu	2x P850	Linux	Unrestricted UNIX Shell Access	7.2	SSH, TELNET, RLOGIN, SHELL
linux3.gl.MYNET.edu	2x P850	Linux	Unrestricted UNIX Shell Access	7.2	SSH, TELNET, RLOGIN, SHELL
titan.MYNET.edu	SGI Challenge XL (20x R10000)	IRIX	Research Computing	6.5.16m	SSH, TELNET, RLOGIN, SHELL
watto.gl.MYNET.edu	Sun Ultra5	Solaris	GL ftp server, AFS/NFS translator	2.6	SSH, FTP, NFS
printhost.MYNET.edu	Sun Ultra5	Solaris	LPRNG Printing Svc.	8	SSH, lpd
cal1.MYNET.edu	Sun NetraX1	Solaris	CorporateTime Calendar	8	SSH
cal2.MYNET.edu	Sun NetraX1	Solaris	CorporateTime Calendar	8	SSH
milter1.MYNET.edu	SunFire 280R	Solaris	Milters: Spam & AntiVirus	8	SSH
milter2.MYNET.edu	SunFire 280R	Solaris	Milters: Spam & AntiVirus	8	SSH

Poncheri, Zachary
(http://MY.NET.9.36)

Device Name (obfuscated)	**Address**	**Function**
www.MYNET.edu	MY.NET.12.11	Web Server (Primary)
cachexxxx.MYNET.edu	MY.NET.16.42	Web Cache
xxxxcampus-private.vpn.MYNET.edu	MY.NET.16.114	VPN
xxxxcampus.vpn.MYNET.edu	MY.NET.16.106	VPN
daxx.MYNET.edu.	MY.NET.8.8	Router
xxxxxcampus-gw.MYNET.edu	MY.NET.8.207	Router
c003xx.MYNET.edu	MY.NET.8.207	Router
Bigxxx-gw.MYNET.edu.	MY.NET.8.2	Router
c003xx.MYNET.edu	MY.NET.21.30	Router
mxinxxx.MYNET.edu	MY.NET.12.6	Mail Server
listxxx.MYNET.edu	MY.NET.24.20	Mail Server
mxXXXin.MYNET.edu	MY.NET.25.70	Mail Server
mxXXXin.MYNET.edu	MY.NET.25.67	Mail Server
mxXXin.MYNET.edu	MY.NET.25.71	Mail Server
mxXXXin.MYNET.edu	MY.NET.25.72	Mail Server
mxXXXin.MYNET.edu	MY.NET.25.73	Mail Server
mxXin.MYNET.edu	MY.NET.25.69	Mail Server
mxXXin.MYNET.edu	MY.NET.25.74	Mail Server
mxXXXin.MYNET.edu	MY.NET.25.68	Mail Server
Alumni.MYNET.edu	MY.NET.60.17	Mail Server
imap.cs.MYNET.edu	MY.NET.34.14	Mail Server
mxXout.MYNET.edu	MY.NET.25.10	Mail Server
mxXXout.MYNET.edu	MY.NET.25.11	Mail Server
mxXXXout.MYNET.edu	MY.NET.25.12	Mail Server
lxx.xxx.MYNET.edu	MY.NET.9.3	Mail Server
mailXX.xx.MYNET.edu	MY.NET.34.5	Mail Server
lanxx.MYNET.edu	MY.NET.30.3	Novell Netware Server
Lanxxx.MYNET.edu	MY.NET.30.4	Novell Netware Server
Ftpxxx.MYNET.edu	MY.NET.16.30	FTP Server
MYNET5.MYNET.edu.	MY.NET.1.5	DNS
MYNET4.MYNET.edu.	MY.NET.1.4	DNS
MYNET3.MYNET.edu.	MY.NET.1.3	DNS
voxx.noc.MYNET.edu.	MY.NET.9.7	Dial-in Server
grxxx.xxx.MYNET.edu.	MY.NET.9.5	Dial-in Server
ciscoxxxx-dw.MYNET.edu.	MY.NET.2.206	Dial-in Server
ciscoxxx-dw.MYNET.edu.	MY.NET.2.205	Dial-in Server
ciscoxx-dw.MYNET.edu.	MY.NET.2.204	Dial-in Server
xxx.xxx.MYNET.edu.	MY.NET.9.12	Authentication Server
anxxx.MYNET.edu.	MY.NET.30.66	Authentication Server

Appendix D: ISC Handler Diary Entries (January 16-24, 2004)

http://isc.sans.org/diary.php?date=2004-01-16
http://isc.sans.org/diary.php?date=2004-01-17
http://isc.sans.org/diary.php?date=2004-01-18
http://isc.sans.org/diary.php?date=2004-01-19
http://isc.sans.org/diary.php?date=2004-01-20
http://isc.sans.org/diary.php?date=2004-01-21
http://isc.sans.org/diary.php?date=2004-01-22
http://isc.sans.org/diary.php?date=2004-01-23
http://isc.sans.org/diary.php?date=2004-01-24

Below are the individual ISC Handler Diary entries for the dates that covered this analysis.

Poncheri, Zachary

Handler's Diary January 16th 2004

Handler on Duty: Johannes Ullrich

Updated January 17th 2004 18:18 UTC

0x01 trojan update (ev1.net host), openssl proof of concept exploit, HP mystery ssh patch

ev1.net trojan (was: Yahoo.fr)

A user submitted a fake e-mail, which is using the %01 MSIE bug to trick the user into downloading a Trojan.

The virus spreading this email is smart enough to tailor the 'From' address to match the users domain. So for example, if your email address is 'user@example.com', the from address will read:
Example.com's Virus Department.

The fake URL will show up as 'http://example.com' followed by the 0x01 character and a randomized URL.

Likely in an effort to dwarf attempts to capture the trojan and shut down the site, the site uses multiple redirects and will only deliver the trojan if the user is using Microsoft Internet Explorer. In order to accomplish this, java script and cgi scripting is used.

The trojan is only delivered once to a given IP address. The final URL used to download the trojan is http:/ /66.98.208.24/cgi-bin/page.cgi at this point, but it has been changing.

The ISP hosting this site, EV1.net, was notified via e-mail to abuse, and replied that the virus has been removed. However, even after this reply was received, the trojan was still accessible via this URL.

A phone call to the customer service department of ev1.net was answered. The ev1.net representative was not able to respond to the case and was not able to provide a phone contact for the ev1.net abuse department.

Later today (early afternoon EST), the host was shut down. Another user reported to us, that a very similar URL was used at ev1.net back in December 2003:

http://66.98.188.67:180/cgi-bin/page.cgi

Back then, the e-mail claimed to include a "Gift Card from Sears".

OpenSSL POC exploit

Exploit code for the older ASN.1 vulnerability in OpenSSL has been posted to various mailing lists. Please double check that your openssl installs are current. Remember, some software may

Poncheri, Zachary
not use the dynamic library. Such software has to be recompiled to link it against the new version.

HP Mystery SSH patch

HP released a patch for ssh on Tru64 Unix. The patch does not state what vulnerability it fixes.

Johannes Ullrich, SANS Inst., jullrich at sans.org

Poncheri, Zachary

Handler's Diary January 17th 2004

Handler on Duty: Davis Ray Sickmon, Jr

Updated January 18th 2004 03:11 UTC

More SoBig comments, and Whack-A-Scam, Ultr@VNC Vulnerability

Alex Shipp of Message Labs email further comments on the SoBig.F resurrection. Alex pointed out that their statistics show no overall increase in SoBig.F emails - instead, just normal fluctuation in the daily statistics.

It's been pointed out that while the trojan-loaded website EV1.NET has been shut down, in typical whack-a-mole fashion, a new one has already popped up at chwolter.com. If you happen to see any more of these pop up, it's probably worth mentioning them.

Ultr@VNC[1] is a VNC variation for administrating Windows based platforms remotely. It supports Windows logins and access rights - however, today Secure Network Operations released a new security escalation example (you have to already be logged into VNC) and Ultr@VNC has not been patched yet to fix the problem. A quick fix (via commenting out some lines and recompiling) was mentioned in the release on BugTraq.

(Mentioned because I know a number of Windows admins who make use of some of the VNC variants for remote server configuration. Since it's unknown when the patch will be released at this time,)

[1]http://ultravnc.sourceforge.net/

Handler On Duty, Davis Ray Sickmon, Jr Midnight Ryder Technologies (http://www.midnightryder.com)

Poncheri, Zachary

Handler's Diary January 18th 2004

Handler on Duty: Tony Carothers

Updated January 18th 2004 19:51 UTC

SPAM-Let the time fit the crime;

Time to speak out & help the Justice System

The US Government is asking for feedback on sentencing guidelines in regards to spammers. With the implementation of the "CAN-SPAM Act of 2003", they are asking the experts for feedback on punishment.

An article by 'The Register' (link shown below) gives a good summary of the Sentencing Guideline.

Link: http://www.ussc.gov/FEDREG/fedr0104.htm
http://www.theregister.co.uk/content/55/34951.html

Poncheri, Zachary

Handler's Diary January 19th 2004

Handler on Duty: Patrick Nolan

Updated January 19th 2004 19:04 UTC

Redhat Kernel Packages (one AMD64 CVE security item), Bagel AV Vendor Summary

"Updated kernel packages available for Red Hat Enterprise Linux 3"

Advisory: RHSA-2004:017-06
"On AMD64 systems, a fix was made to the eflags checking in 32-bit ptrace emulation that could have allowed local users to elevate their privileges. The Common Vulnerabilities and Exposures project (cve.mitre.org) has assigned the name CAN-2004-0001 to this issue."
http://rhn.redhat.com/errata/RHSA-2004-017.html

Affected Products:
Red Hat Enterprise Linux AS (v. 3)
Red Hat Enterprise Linux ES (v. 3)
Red Hat Enterprise Linux WS (v. 3)

CVEs (cve.mitre.org): CAN-2004-0001
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2004-0001

Bagel AV Vendor Summary

Reports to the ISC indicate that AV gateways intercepting this worm and configured to "Autoreply" to the spoofed "From:" source are once again causing needless congestion (see SOBIG issues). Offenders should consider changing this configuration.

Three write-ups specify the worm's email will have an attachment "Length: 15,872 bytes" and one write-up says it is "an .exe file extension and consists of 3 - 11 randomly-generated lowercase characters."

After infection and initiation of it's email routine AV write-ups state that Bagel "will initialize and open a TCP socket in listening mode on port 6777."

The Trojan Retrieval Routine consists of:

"[HTTP connection]
HTTP GET REQUEST
GET /1.php?p=6777&id=[uid value, same value as used in the registry key]
User-Agent: beagle_beagle"

Poncheri, Zachary
In AV Vendor write-ups so far the worm has hardcoded URLS which have not had 1.php available.

One Vendor (TrendMicro) cryptically reports "This worm may perform port scanning to connect to a remote system."

http://securityresponse.symantec.com/avcenter/venc/data/w32.beagle.a@mm.html
http://vil.nai.com/vil/content/v_100965.htm
http://www3.ca.com/virusinfo/virus.aspx?ID=38019
http://www.sophos.com/virusinfo/analyses/w32baglea.html
http://www.f-prot.com/virusinfo/descriptions/bagle_a.html
http://www.messagelabs.com/viruseye/threats/list/default.asp
http://wtc.trendmicro.com/wtc/summary.asp

Patrick Nolan

Poncheri, Zachary

Handler's Diary January 20th 2004

Handler on Duty: Tom Liston

Updated January 21st 2004 01:38 UTC

ICMP Echo/HTTP Pattern, HP Mystery Patch Explained, DNS Reflector Attack(?)

Combined ICMP Echo Request and TCP Port 80 Traffic
We have received reports of an odd traffic pattern: a single ICMP echo request followed immediately by an HTTP request for the default website page. This pattern is repeated at a daily rate of approximately 1200 times per day, each sourced from a different IP.

We're "fishing" (rather than "phishing") for information on this. If anyone out there is experiencing the same phenomenon, please drop us a note:

http://isc.sans.org/contact.html

HP Patch Mystery Explained
In the January 16th Diary (http://isc.sans.org/diary.html?date=2004-01-16), we mentioned that HP had made a "mystery" patch available for SSH on Tru64 Unix. This article explains its purpose:

http://news.zdnet.co.uk/software/linuxunix/0,39020390,39119149,00.htm

The patch fixes flaws in both SSH and VPN on Tru64 Unix. The flaws are believed to be present only in the Tru64 versions of these services.

Looking For Signs of Large Scale DNS Reflector Attack
We have received reports of DNS servers suddenly attempting to repeatedly and rapidly resolve a single hostname.

Again, we're on a "fishing" expedition here, folks. Please take a look for this behavior on your networks and report anything you find to us.

http://isc.sans.org/contact.html

Handler on Duty: Tom Liston (http://www.labreatechnologies.com)

Poncheri, Zachary

Handler's Diary January 21st 2004

Handler on Duty: Deb Hale

Updated January 22nd 2004 04:19 UTC

Another Active Day

The Beagle/Bagel has been busy today.

Early this morning Symantec raised it to a level 3 due to the number reported to be out in the wild. They have now posted a removal tool on the web site.
http://securityresponse.symantec.com/avcenter/venc/data/w32.beagle.a@mm.html

Strange Port Activity
Still receiving reports of unusual activity on Ports 80 (Code Red II ?) and 53 (DNS), as well as a continued increase in port 6129, Dameware.
The ISC would like to encourage anyone seeing unusual activity to contact us and let us know what you are seeing.
http://isc.sans.org/contact.html

Deb Hale BCP Enterprise Inc

Poncheri, Zachary

Handler's Diary January 22nd 2004

Handler on Duty: Pedro Bueno

Updated January 22nd 2004 21:20 UTC

Dameware Traffic and mailbag

Dameware Traffic

In yesterdays diary (http://isc.sans.org/diary.html? date=2004-01-21) , we ask you info about 6129 traffic.

Thanks for all the logs sent to us. We are still interested in it if you have full tcpdump packet captures.

In despite of the high number of reports received, until this moment there is no evidence that the 6129 traffic is caused by a Worm. The relevant factor is the low/stable number of sources. (http://www.dshield.org/port_report.php?port=6129&recax=1&tarax=2&srcax=2&percent=N&days=40). We are noticing an interesting pattern in the scanning tool that, apparently, is behind this traffic. The Incident Handler Donald Smith pointed that "it increments the 3rd octet. That will move it cross networks in most cases! So sequential packets might not trigger a scan if you are only counting packets per second to your network"

If you want to participate in the internet storm center, as well as get reports, fight back, and other benefits, we would like to you to consider the use of Dshield, as well its clients to send the logs to Dshield (http://www.dshield.org/howto.php).

Mailbag
We received an email about a possible Nachi/Blaster worm infection in a XP computer. SANS released a very good document about Windows XP security called Windows XP Surviving the first day (http://www.sans.org/rr/papers/index.php?id=1298)

Handler on Duty: Pedro Bueno

Poncheri, Zachary

Handler's Diary January 23rd 2004

Handler on Duty: Jim Clausing

Updated January 24th 2004 00:28 UTC

Updated: Security bulletins from Sun, more Dameware

2 Sun security bulletins
Yesterday, Sun released several security bulletins, we'd like to mention 2 of them here today. The first involves the possibility of a local user being able to gain additional privileges through the loading of arbitrary kernel modules. Sun has released kernel patches for Sun OS 5.7, 5.8, and 5.9 (aka Solaris 7, Solaris 8, and Solaris 9) to address the situation. The second bulletin we'd like to mention addresses a buffer overflow leading to possible remote denial of service or unauthorized root acces against 5.9 (Solaris 9) systems running in.iked (IKE stands for Internet Key Exchange). This vulnerability is apparently in ASN.1 parsing code that Sun uses from SSH, Inc. ASN.1 vulnerabilities were the subject of Cert Advisory CA-2003-26.
You can see the bulletins here:
http://sunsolve.sun.com/private-cgi/retrieve.pl?doc=salert%2F57479&zone_32=category%3Asecurity
http://sunsolve.sun.com/private-cgi/retrieve.pl?doc=salert%2F57472&zone_32=category%3Asecurity

Continuing Dameware traffic
We continue to see a great deal of traffic on port 6129 including new reports of systems being exploited running versions of Dameware that were not supposed to be vulnerable to the previously reported problems. We'll continue to monitor the situation.

Other ports on the rise
We are seeing increases in apparent DNS attacks, and in port 901 and port 2234 traffic. If you have any packet captures of any of this traffic, we would be very interested in taking a look at it, send it to us at http://isc.sans.org/contact.html

FDIC phishing scam
Finally, a report late today of another phishing scam, this one telling people that the Department of Homeland Security has instructed the FDIC to deny federal deposit insurance due to suspected violations of the USA PATRIOT Act. FDIC (the agency that insures bank accounts in the US), has posted a response.http://www.fdic.gov/news/news/SpecialAlert/2004/sa0504.html

--Jim Clausing

Poncheri, Zachary

Handler's Diary January 24th 2004

Handler on Duty: Tan Koon Yaw

Updated January 24th 2004 18:06 UTC

Port 1070, Dumaru Worm, Email Disguised as Microsoft Patch

Port 1070

We received a report that there is an increase scan on port 1070.

If you see any unusual activities or have any sample logs, please let us know.

http://isc.sans.org/port_details.html?port=1070

Dumaru Worm

There is a new variant of worm that sends an attachment as a zip file which contains the worm executable, myphoto.jpg<56 spaces>.exe.

On infected system, it may open a backdoor on port 10000 which allow the attacker to connect and perform malicious actions.

If you have a copy of the worm, please let us know.

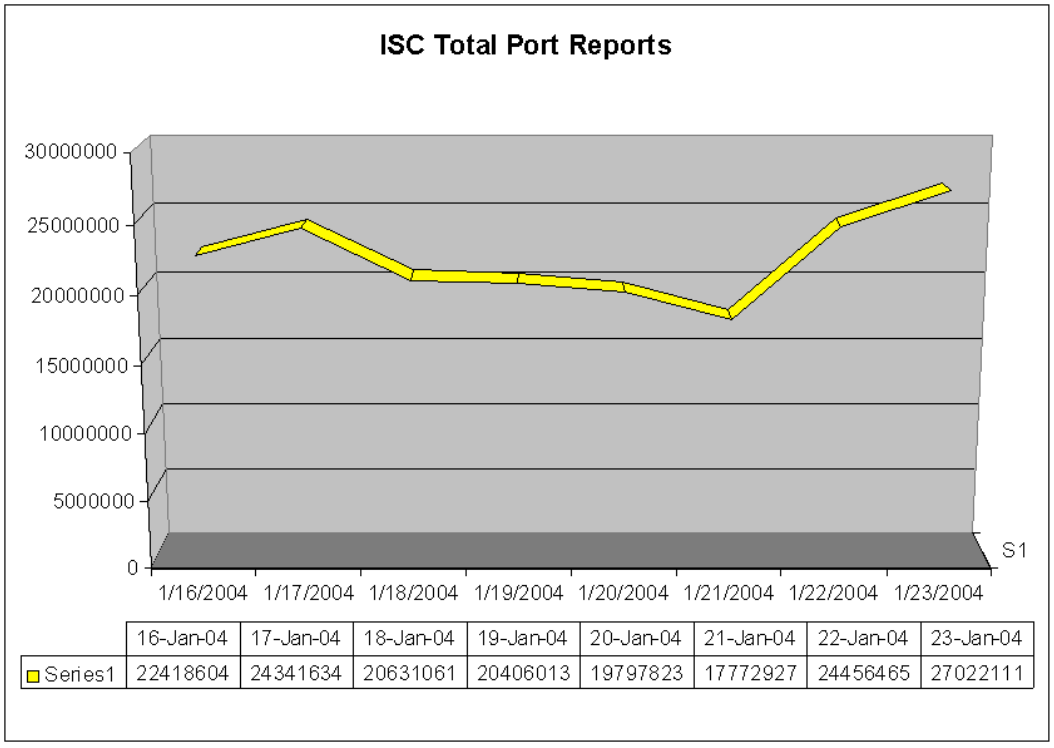
http://securityresponse.symantec.com/avcenter/venc/data/w32.dumaru.y@mm.html
http://www.f-secure.com/v-descs/dumaru_y.shtml
http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_DUMARU.Y
http://www.messagelabs.com/viruseye/info/default.asp?frompage=threats+list&fromURL=%2Fviruseye%2Fthreats%2Flist%2Fdefault%2Easp&virusname=W32%2FDumaru%2EY%2Dmm

Email Disguised as Microsoft Patch

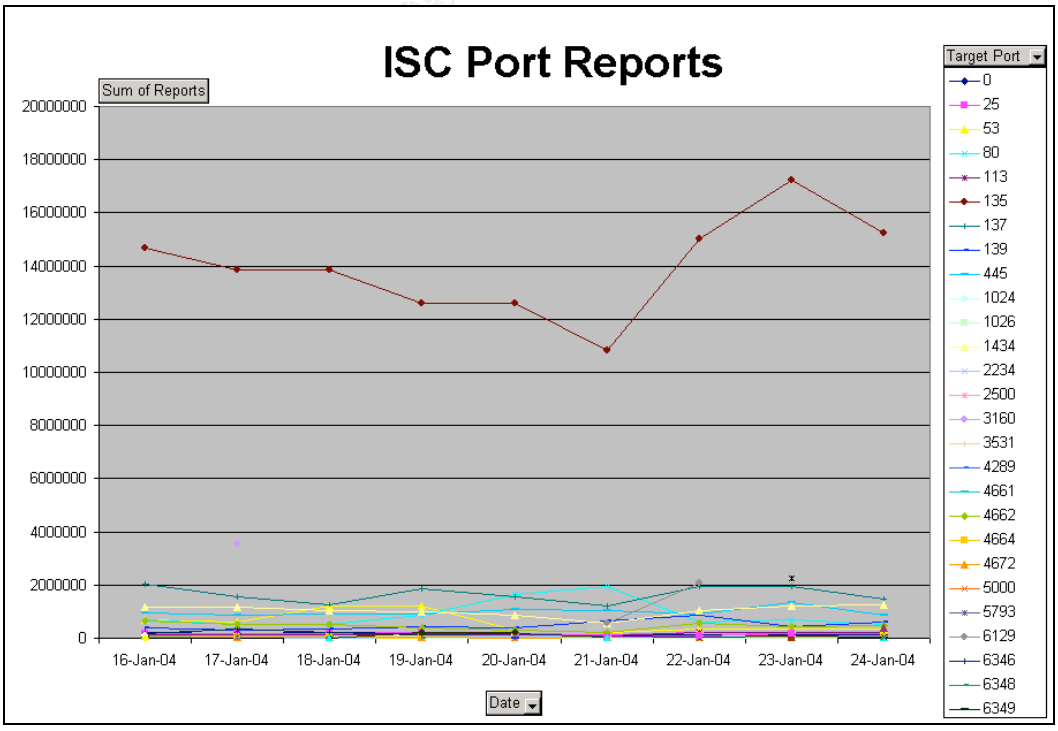
We also received a report on an email disguising as Microsoft Security Patch. According to Microsoft, they will not send patches via email. If you receive such emails, be wary as most likely it is attempting to trick you to execute some malware.

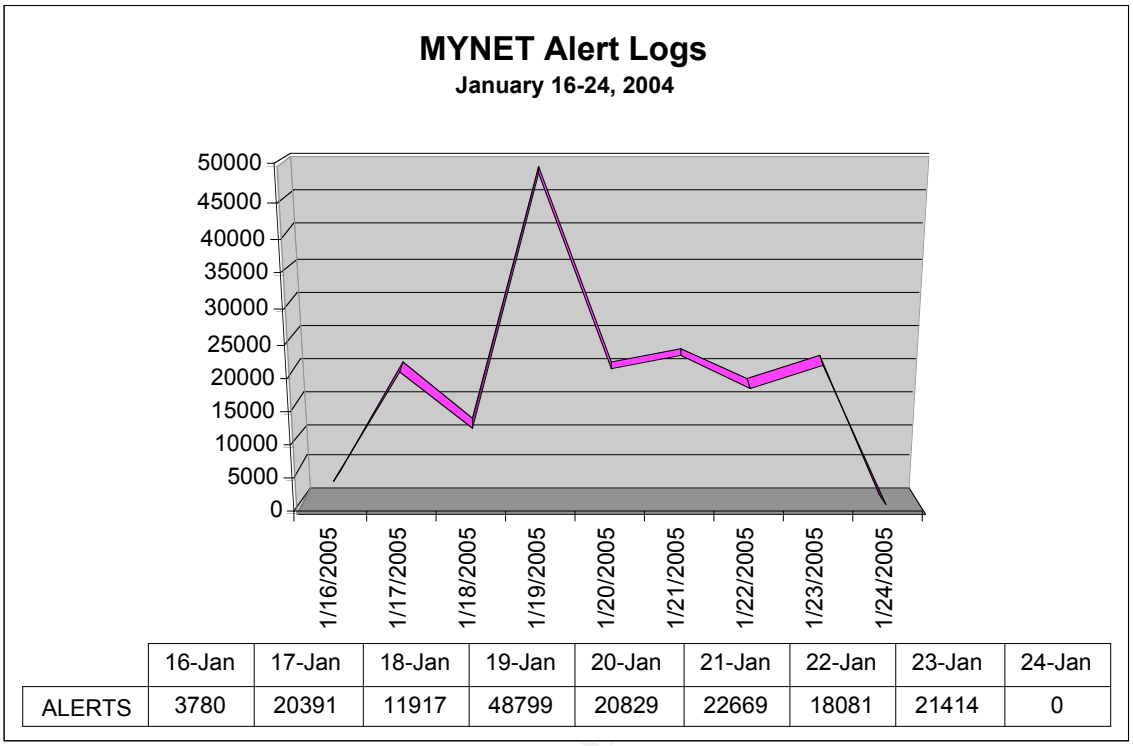
Appendix D: ISC and MYNET Reports

Using data from the Internet Storm Center (isc.sans.org), these Excel spread sheets are compiled to represent the data in hopes of finding trends to correlate with MY.NET.

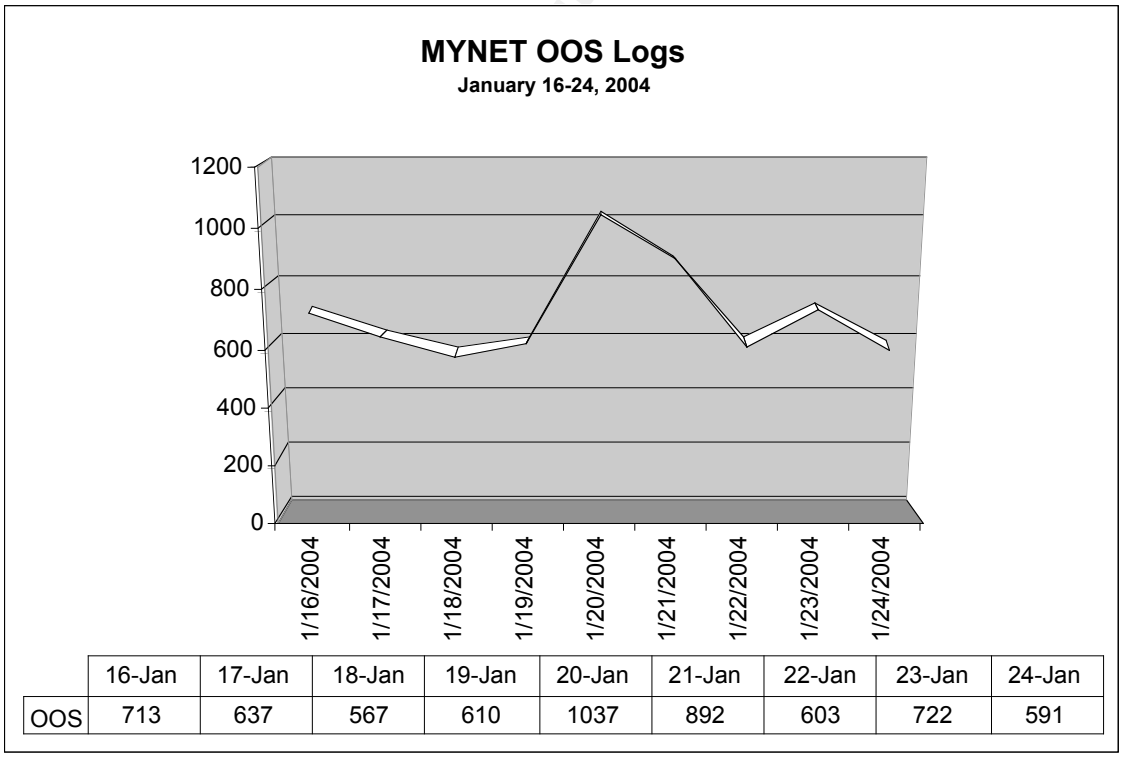


	16-Jan-04	17-Jan-04	18-Jan-04	19-Jan-04	20-Jan-04	21-Jan-04	22-Jan-04	23-Jan-04
Series1	22418604	24341634	20631061	20406013	19797823	17772927	24456465	27022111

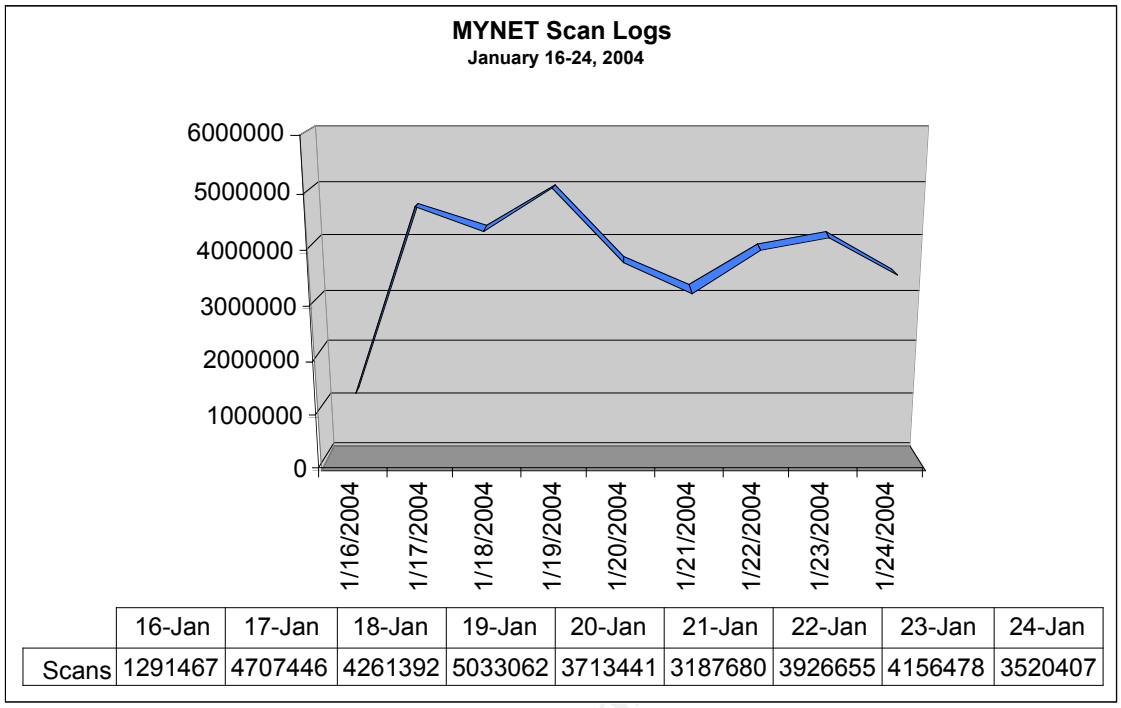




	16-Jan	17-Jan	18-Jan	19-Jan	20-Jan	21-Jan	22-Jan	23-Jan	24-Jan
ALERTS	3780	20391	11917	48799	20829	22669	18081	21414	0



	16-Jan	17-Jan	18-Jan	19-Jan	20-Jan	21-Jan	22-Jan	23-Jan	24-Jan
OOS	713	637	567	610	1037	892	603	722	591



	16-Jan	17-Jan	18-Jan	19-Jan	20-Jan	21-Jan	22-Jan	23-Jan	24-Jan
Scans	1291467	4707446	4261392	5033062	3713441	3187680	3926655	4156478	3520407

Appendix E: Vulnerability Reference Reports

Name	CVE-1999-0189
Description	*Description: Solaris rpcbind listens on a high numbered UDP port, which may not be filtered since the standard port number is 111.*
URL	http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0189

Name	rpc-32771 (330) RPC bind service on improper port
Description	*Normally, the rpcbind service only listens on port 111. Under Solaris, the rpcbind service also listens under port 32771, which sometimes allows attackers to bypass packet filtering.*
URL	http://xforce.iss.net/xforce/xfdb/330

Name	bugtraq-id 205
Description	*The rpcbind program that converts RPC program numbers into universal addresses. When a client makes an RPC call to a given program number, it first connects to rpcbind on the target system to determine the address where the RPC request should be sent. Under Solaris 2.x rpcbind not only listens on the TCP / UDP port 111, but it also listens on UDP ports greater than 32770. The exact number is dependent on the OS release and architecture. Thus, packet filtering devices that are configured to block access to rpcbind / portmapper, may be subverted by sending UDP requests to rpcbind listening above port 32770.This vulnerability may allow an unauthorized user to obtain remote RPC information from a remote system even if port 111 is being blocked.*
URL	http://www.securityfocus.com/bid/0205

Name	snort-id 205
Description	*The portmapper service registers all RPC services on UNIX hosts. It can be queried for all RPC services running, the RPC program name and version, the protocol (TCP or UDP), and the port where the service listens. This can provide an attacker with valuable information about what RPC services are offered and on which ports.*
URL	http://www.snort.org/snort-db/sid.html?sid=599

http://www.securityfocus.com/bid/0205

bugtraq id **205**
object rpcbind
class Design Error
cve CVE-1999-0189
remote Yes
local Yes
published Jun 04, 1997
updated Jun 01, 1999
vulnerable Sun Solaris 2.3
Sun Solaris 2.4 _x86
Sun Solaris 2.4
Sun Solaris 2.5 _x86
Sun Solaris 2.5
Sun Solaris 2.5.1 _x86
Sun Solaris 2.5.1 _ppc
Sun Solaris 2.5.1
Wietse Venema Rpcbind Replacement 2.0
- Sun Solaris 2.4
- Sun Solaris 2.4 _x86
- Sun Solaris 2.5
- Sun Solaris 2.5 _x86
- Sun Solaris 2.5.1
- Sun Solaris 2.5.1 _ppc
- Sun Solaris 2.5.1 _x86

not vulnerable FreeBSD FreeBSD 3.3
SSH Communications Security SSH 1.2.27
- Debian Linux 2.2
- Debian Linux 2.2 68k
- Debian Linux 2.2 alpha
- Debian Linux 2.2 arm
- Debian Linux 2.2 powerpc
- Debian Linux 2.2 sparc

Sun Solaris 2.6 _x86
Sun Solaris 2.6
Wietse Venema Rpcbind Replacement 2.1
- Sun Solaris 2.4
- Sun Solaris 2.4 _x86
- Sun Solaris 2.5
- Sun Solaris 2.5 _x86
- Sun Solaris 2.5.1
- Sun Solaris 2.5.1 _ppc
- Sun Solaris 2.5.1 _x86
- Sun Solaris 2.6
- Sun Solaris 2.6 _x86

Last Updated: July 28th, 2017

Upcoming Training

SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Boston 2017	Boston, MA	Aug 07, 2017 - Aug 12, 2017	Live Event
SANS Adelaide 2017	Adelaide, Australia	Aug 21, 2017 - Aug 26, 2017	Live Event
SANS Network Security 2017	Las Vegas, NV	Sep 10, 2017 - Sep 17, 2017	Live Event
SANS vLive - SEC503: Intrusion Detection In-Depth	SEC503 - 201709,	Sep 11, 2017 - Oct 18, 2017	vLive
Baltimore Fall 2017 - SEC503: Intrusion Detection In-Depth	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	vLive
SANS London September 2017	London, United Kingdom	Sep 25, 2017 - Sep 30, 2017	Live Event
SANS Baltimore Fall 2017	Baltimore, MD	Sep 25, 2017 - Sep 30, 2017	Live Event
Community SANS Scottsdale SEC503	Scottsdale, AZ	Oct 02, 2017 - Oct 07, 2017	Community SANS
SANS October Singapore 2017	Singapore, Singapore	Oct 09, 2017 - Oct 28, 2017	Live Event
Community SANS Ottawa SEC503	Ottawa, ON	Oct 16, 2017 - Oct 21, 2017	Community SANS
SANS Berlin 2017	Berlin, Germany	Oct 23, 2017 - Oct 28, 2017	Live Event
San Diego Fall 2017 - SEC503: Intrusion Detection In-Depth	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	vLive
SANS San Diego 2017	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	Live Event
SANS Seattle 2017	Seattle, WA	Oct 30, 2017 - Nov 04, 2017	Live Event
SANS Paris November 2017	Paris, France	Nov 13, 2017 - Nov 18, 2017	Live Event
Community SANS Pensacola SEC503	Pensacola, FL	Nov 27, 2017 - Dec 02, 2017	Community SANS
SIEM & Tactical Analytics Summit & Training	Scottsdale, AZ	Nov 28, 2017 - Dec 05, 2017	Live Event
SANS Cyber Defense Initiative 2017	Washington, DC	Dec 12, 2017 - Dec 19, 2017	Live Event
SANS Security East 2018	New Orleans, LA	Jan 08, 2018 - Jan 13, 2018	Live Event
SANS Las Vegas 2018	Las Vegas, NV	Jan 28, 2018 - Feb 02, 2018	Live Event
SANS Dallas 2018	Dallas, TX	Feb 19, 2018 - Feb 24, 2018	Live Event
SANS OnDemand	Online	Anytime	Self Paced
SANS SelfStudy	Books & MP3s Only	Anytime	Self Paced