Global Information Assurance Certification Paper

GCIA Practical Assignment v3.0
SANS Peachtree 2002
May 10, 2002

Paul J. Farley Jr

Table of Contents

If you talk with five people you will get five different answers to the question "Why is correlation important in intrusion detection"? The reason for this is since each of us has our own experiences guiding our perceptions; each will look at a given set of facts slightly differently. This can result in great differences in the interpretation of those facts, and so vary the actions taken to mitigate the risk of a perceived threat. Merriam-Webster's dictionary defines correlation as:

> "A relation existing between phenomena or things or between mathematical or statistical variables which tend to vary, be associated, or occur together in a way not expected on the basis of chance alone; the act of correlating."

The value of correlation seems to be frequently overlooked or at least not explored for all possible avenues. In intrusion detection, when anomalous traffic is observed, often more questions than answers come to mind. Questions such as: "Is this an attack or a response to something I did? Who does this IP belong to? Are they targeting me? Am I vulnerable? Did they succeed in the attack"? Many times simply by observing the traffic all of these questions cannot be answered. Correlation can come from many sources, inside and outside your organization. The more correlation you can perform, the more confident you can be your analysis is correct. The information that can be obtained via correlation can completely change the analysis of an event, making it more or less severe. The majority of the questions deal with the business impact of the technical event, rather than the technical event itself. Therein lays the most important aspect of any technical discipline, an understanding of how the technical minutiae affect your business. Correlation gives the supporting evidence to enable appropriate actions to be taken according to an organization's security policies and the law.

We will explore both technical and business correlation, and explain the impact of both of these in the larger picture of protecting against unauthorized use of resources and information.

Types of Intrusion Detection systems:

There are different types of intrusion detection, defined as follows:

Network Based Intrusion Detection (NIDS)

Network Intrusion Detection Systems are centered around the inspection and monitoring of network traffic. These systems examine network traffic for signs of anomalous traffic and provide some alert, logging or action. The anomalous traffic is identified through the use of rules that define such traffic. When the system matches a rule, it creates an event. Some NIDS

are capable of taking action such as responding to the traffic to break the session, some can execute pre-defined actions, others can only log.

Systems Integrity Verifiers (SIV)

Systems Integrity Verifiers provide a host-centric view of any changes to key files and file systems. These systems generate a baseline of the file systems they are monitoring, and when there are changes they provide notification. Typically, the most critical files that effect system stability do not change unless the system is being patched or upgraded. A notification at this time would be a non-event since the systems administrator would be authorized to make that change. Notification of those changes at any other time however, could indicate that an unauthorized person was attempting to place a backdoor in the system or just bring the system down. These systems are usually considered a second line of defense, since the files or file systems have to be reached in order to cause it to generate an event. Some of these systems can only log, others can do more complex event handling.

Log File Monitors (LFM)

Log file monitors are used to monitor systems which handle events by writing them to a log. Log File Monitors look for defined patterns in the log files, and when the pattern is matched an event is generated. Some examples of systems that can be monitored in such a manner are web servers, NT event logs, syslog and firewalls. Many of these tools come pre-defined for the most common formats of log files, and some come with the additional ability to define custom formats. This is particularly useful for monitoring home grown applications, and legacy applications that might still be in use.

Types of Correlation

There are several different layers of correlation that can occur. In general, each level of correlation takes the output of the previous level as at least one of the inputs. The lower layers are more technically focused, and as you escalate through each layer, more external and non-technical input is used. This supports the ultimate goal of correlation, of tying a technical event to its impact on your business. Not all events go through all the layers, some are either known or are correlated sufficiently that further analysis is not warranted. We will explore each of these layers.

Basic Correlation – This correlation occurs within the data generated by an IDS component. An example of this would be using a tool to summarize and "clean-up" the output of the events to show the most important information. This layer brings attention to the events that need further consideration. For example, at this layer the mechanism and order of an attack can be seen, since all the events relating to a particular source IP address are shown, and it can indicate if they were doing port scans,

host scans or a series of attacks against a host or hosts. Such information might not be easily visualized by manually reviewing the raw IDS output, or looking through web server logs.

Intermediate Correlation – This correlation occurs between events from a singular type of source, with multiple instances in the environment. This layer provides information regarding the magnitude and possible effectiveness of an attack. An example of this would be correlation between two NIDSs. Depending on the placement, such correlation might show that an attack that was seen outside the firewall (externally placed NIDS) was inside the network (internally placed NIDS). This would indicate a configuration problem or vulnerability, and need to be addressed immediately. A LFM might show that multiple web servers are the targets of the same type of attack, however if some of them are not vulnerable to that attack (i.e. an Apache web server logging IIS exploit attacks) then that would indicate the level of knowledge of your network the attack has, and possibly the skill level of the attacker. A critical factor in being able to correlate events from multiple devices is to ensure those devices are time synchronized. This is important not just for ease of analysis, but also if the events are to be used as evidence in a court of law, lack of time synchronization will negatively impact the viability of the evidence. Time synchronization is most important beginning in this layer and above.

Advanced Internal Correlation – This correlation occurs between events from multiple sources and typically begins to apply some business intelligence to the correlation. An example of this would be a correlation between an event from the NIDS showing a buffer overflow attack on the web server, and a log file event from the web server showing that request was refused. This layer of correlation can show the success and or extent of the penetration. This is particularly true if the NIDS detects an attack, which shows in the log file of the server attacked, and a SIV shows that files have been changed. Frequently this level of correlation is performed manually by a security analyst, since even with an enterprise security event management tool there are data normalization issues with correlating disparate events from multiple tools and platforms.

Advanced External Correlation – Correlation to events outside your organization would occur at this layer, as the knowledge that others are (or are not) being hit with similar events has a direct impact on the severity of the correlation. This is part of ensuring that a complete a picture of the intent, methods and achievements of an attacker are systematically evaluated. Most events that reach this level will result in either a technical response to mitigate future attacks such as changing network configuration or firewall rules; or a non-response when it is determined that an event does not warrant action after it has been sufficiently investigated.

Business Correlation – This correlation comes to factors outside of technical boundaries such as knowledge of any competitors, employment actions, economic factors, strategic plans, terrorist threats, classified projects, pending sales and other market conditions that would be pertinent and affect the severity of the event. This cannot be automated since it is not systematic, and it requires as input events that have been through all of the previous stages of correlation. This correlation should be completed in a joint effort between technical and business analysts, after the other levels of correlation indicate it is a high enough severity to warrant the escalation. In order to avoid a loss of credibility for the process and the security

department in the eyes of executive management, security personnel should be judicious with events that are allowed to reach this level. Lower severity events may be summarized in a periodic report to management to ensure there is visibility into the events that are being handled, without causing unnecessary alarm.

Each layer of correlation is important, and yields information that can be used to mitigate future risk from a like attack. The higher the level of correlation, the more intensive the resource demand becomes, so it is prudent to keep correlation to the lowest level possible to enable an appropriate response.

Summary

Correlation is an integral part of effective intrusion detection, as it can shed more light on a discrete event than can be determined by the isolated evaluation of just that event. Since intrusion analysts are not in the minds of the attackers, the more perspectives on the attack they can view, the more complete the picture of the attack and therefore the more appropriate defense or action in response. A formal, defined correlation process ensures that every event of interest is evaluated appropriately, and minimizes the risk of mis-categorizing an event's severity and therefore recommending inappropriate action (or non-action). Over time, if discipline is maintained, both the process of intrusion detection and the intrusion analyst will gain credibility in the organization. This credibility will ultimately have a positive effect on the overall security of their organization.

References:

1. "Event Correlation: The enabler of Active Internet Security Management",(2001),URL: http://www.open.com/pdf/eventcorr.qxd.pdf ,(10 May 2002).

2. Walker, John Q., Ph.D.,(2002), "Security Event Correlation: Where Are We Now?", URL: http://www.netiq.com/Downloads/Library/white_papers/Security_Event_Correlation-Where_Are_We_Now.pdf ,(10 May 2002).
3. Jordan, Chris, "Analyzing IDS Data", (30 May 2000), http://online.securityfocus.com/infocus/1201

4. Rasmussen, Scott, "Centralized Network Security Management: Combining Defense in Depth with Manageable Security", (20 January 2002), URL: http://rr.sans.org/practice/central_netsec.php ,(10 May 2002).

5. Buecker, Axel, & Edwards, David, "Recommended Best Practices for Risk Management with Tivoli Risk Manager", (28 Feb. 2002), URL: www.redbooks.ibm.com/redpapers/abstracts/redp0202.html ,(10 May 2002).

6. Merriam-Webster's Collegiate Dictionary, URL: http://www.m-w.com (10 May 2002).

Assignment #2 – Network Detects

Introduction

One of my customers allowed me to use their network to collect the detects in assignment two. In exchange for their generosity I agreed to provide them a full analysis when I completed the practical. I placed a Mandrake Linux 8.1 box with tcpdump 3.6.2 onsite for just over a week. It was placed between their border router and their firewall, to be able to see all the traffic coming to them. Both their border router and firewall are externally managed, and their web presence is hosted offsite. Tcpdump was configured to use a snap length of 150, for disk space considerations. As such there is an understood tradeoff between being able to see the entire payload of anomalous traffic and running out of disk space.

For correlation I used as many sources as I could find that would provide information about the attack or the source IP address. The usefulness of this correlation will be discussed further in the white paper section of the practical. To ensure the reader understands some of the sources however, I will briefly explain them here.

Incidents.org (http://www.incidents.org) - Most readers of this practical will recognize the usefulness of the SANS family of websites in gathering information about an attack or attacker. In particular I found the "IP Information" and "Port Report" functions most helpful in finding if others were experiencing similar activity.

MyNetWatchman.com - MyNetWatchman uses automated agents deployed at many different locations around the world. The agents watch firewall logs for anomalous entries. The agents then send the entries of interest to a central MyNetWatchman analysis server. MyNetWatchman then aggregates the entries into incidents and sends them to the ISP of record for the offending IP address (see Figure 1). The primary goal of MyNetWatchman is providing a system where a compromised host's owner can be notified, and that machine secured, improving the overall security for all concerned. Since MyNetWatchman is an aggregation point for such a distributed system of sensors, it provides a good location to research if an IP is a known offender. To learn more about MyNetWatchman and get a more in-depth explanation you can visit: http://www.myNetWatchman.com.

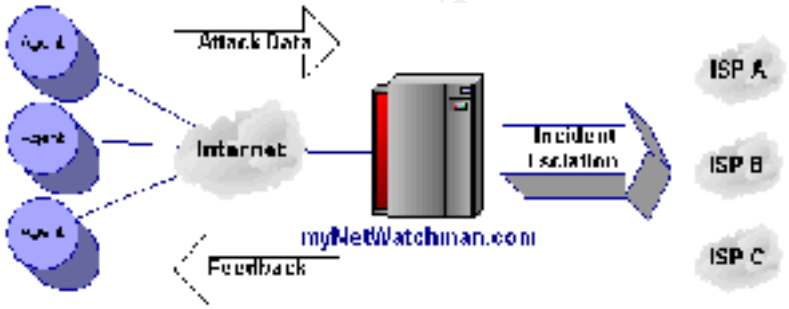


Figure 1 – myNetWatchman Architecture
http://www.mynetwatchman.com/vision.htm

Network Detect #1 - sadmind Worm

```
[**] [1:1375:2] WEB-MISC sadmind worm access [**]
[Classification: Attempted Information Leak] [Priority: 2]
02/16-02:41:08.970194 218.7.3.19:43483 -> MY.NET.CLASSC.170:80
TCP TTL:234 TOS:0x0 ID:37937 IpLen:20 DgmLen:58 DF
***AP*** Seq: 0xAD916A18 Ack: 0x23E4A13F Win: 0x2238  TcpLen: 20
[Xref => http://www.cert.org/advisories/CA-2001-11.html]

[**] [1:1375:2] WEB-MISC sadmind worm access [**]
[Classification: Attempted Information Leak] [Priority: 2]
02/16-02:41:09.081399 218.7.3.19:43483 -> MY.NET.CLASSC.170:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:58
***AP*** Seq: 0x186A91AD Ack: 0x186A91AD Win: 0x445E  TcpLen: 20
[Xref => http://www.cert.org/advisories/CA-2001-11.html]
```

Source of Trace:

My customer's network.

Detect Generated By:

Snort v 1.83 with the following rule:

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-MISC sadmind worm access"; flags:A+; content:"GET x HTTP/1.0"; offset:0; depth:15; classtype:attempted-recon; reference:url,www.cert.org/advisories/CA-2001-11.html; sid:1375; rev:2;)

Probability the Source Address was Spoofed:

Low to nil, since the source and destination hosts established a conversation as part of this attack.

02:41:08.630141 218.7.3.19.43483 > MY.NET.CLASSC.170.80: S 2911988247:2911988247(0) win 8760 <mss 1460> (DF)
02:41:08.631806 MY.NET.CLASSC.170.80 > 218.7.3.19.43483: S 602186046:602186046(0) ack 2911988248 win 17520 <mss 1460> (DF)
02:41:08.924693 218.7.3.19.43483 > MY.NET.CLASSC.170.80: . ack 1 win 8760 (DF)
02:41:08.970194 218.7.3.19.43483 > MY.NET.CLASSC.170.80: P 1:19(18) ack 1 win 8760 (DF)
02:41:09.081275 MY.NET.CLASSC.170.80 > 218.7.3.19.43483: P 1:225(224) ack 19 win 17502 (DF)
02:41:09.081399 MY.NET.CLASSC.170.80 > 218.7.3.19.43483: F 225:225(0) ack 19 win 17502 (DF)
02:41:09.382843 218.7.3.19.43483 > MY.NET.CLASSC.170.80: . ack 225 win 8760 (DF)
02:41:09.384184 218.7.3.19.43483 > MY.NET.CLASSC.170.80: . ack 226 win 8760 (DF)
02:41:09.499727 218.7.3.19.43483 > MY.NET.CLASSC.170.80: F 19:19(0) ack 226 win 8760 (DF)
02:41:09.500996 MY.NET.CLASSC.170.80 > 218.7.3.19.43483: . ack 20 win 17502 (DF)

Description of the Attack:

The sadmind/IIS worm attempts to use a vulnerability in Solaris systems to compromise those systems and use them to attack other Solaris and IIS systems. The exploit against an IIS system is that of a directory transversal attack.

Attack Mechanism:

Phillip Cherbaka's GCIH Practical does an excellent job outlining the attack mechanism of this worm. In terms of the IIS portion of the attack, the goal is to pass Unicode characters equivalent to the "\" character to move out of the directory tree for the web server and affect system files. If the attack is successful, code can be executed as the "IUSR_MachineName" account, which is the account the IIS service runs as on Windows NT and Windows 2000 machines. This code could really be anything found on the IIS system, including executing something such as tftp.exe (which is present by default on Windows NT and Windows 2000 systems) and use that to download more exploit code of whatever flavor desired.

Correlations:

Two locations confirm this is a known “bad guy”, showing multiple scans across many hosts.

Dshield.org: http://www.dshield.org/ipinfo.php?ip=218.7.3.19
Shows 6798 records of this IP address against 2862 different addresses.

myNetWatchman:
The correlation from MyNetWatchman.com shows this IP address running various scans across a number of IP address ranges.
The table of agent reports was truncated for brevity, but this incident goes back with at least one scan per day to 02 Jan 2002.
The full incident log from MyNetWatchman can be accessed via : http://www.myNetWatchman.com/LID.asp?IID=2491282

Incident Id : 2491282	Source Ip : 218.7.3.19
Source Name :	Provider Domain : hr.hl.cn
Net Bios Name :	DNS Name :
Total Event Count : 493	Total Distinct Agent : 160/70450
Response : No Recent Activity	
Status Description : Closed	
Exclusion Reason :	

Network Name/NextNIC	Start IP - End IP
APNIC4/APNIC	218.0.0.0 - 218.255.255.255
APNIC-AP/DUMMY	218.0.0.0 - 218.255.255.255
CHINANET-CN/DUMMY	218.0.0.0 - 218.31.255.255
CHINANET-HL/DUMMY	218.7.0.0 - 218.10.255.255

```
NextNIC:99999
Whois provider: hr.hl.cn
```

```
% Rights restricted by copyright. See http://www.apnic.net/db/dbcopyright.html
% (whois7.apnic.net)

inetnum:     218.7.0.0 - 218.10.255.255
netname:     CHINANET-HL
descr:       CHINANET heilongjiang province network
descr:       China Telecom
descr:       A12,Xin-Jie-Kou-Wai Street
descr:       Beijing 100088
country:     CN
```

```
admin-c:     CH93-AP
tech-c:      CX58-AP
mnt-by:      MAINT-CHINANET
mnt-lower:   MAINT-CHINANET-HL
changed:     hostmaster@ns.chinanet.cn.net 20010510
source:      APNIC

person:      Chinanet Hostmaster
address:     A12,Xin-Jie-Kou-Wai Street
country:     CN
phone:       +86-10-62370437
fax-no:      +86-10-62053995
e-mail:      hostmaster@ns.chinanet.cn.net
nic-hdl:     CH93-AP
mnt-by:      MAINT-CHINANET
changed:     hostmaster@ns.chinanet.cn.net 20000101
source:      APNIC

person:      CHE XUESONG
address:     HEILONGJIANG Province liu chuansen
country:     CN
phone:       +86-0451-5630553
fax-no:      +86-0451-5630553
e-mail:      chexs@public.hr.hl.cn
nic-hdl:     CX58-AP
mnt-by:      MAINT-CHINANET-HL
changed:     chexs@public.hr.hl.cn 20000804
source:      APNIC
```

Most Recent Event Date/Time (UTC)	Agent Alias	Agent Type	Log Type	Target Ip	# of IPs Targeted	IP Protocol	Target Port	Port/ Issue Description	Source Port	Explanation	Event Count
16 Feb 2002 13:11:40	cihm	win32	Linksys	24.47.x.x	1	6	111	Remote Procedure Call RPC Exploits	52700	mNW Info	1
16 Feb 2002 01:29:55	nozero	win32	BlackICE	12.238.x.x	1	6	80	HTTP HTTP port probe	51431	advICE | mNW Info	1
15 Feb 2002 23:44:19	-Fred-	win32	Zone Alarm	64.194.x.x	1	6	111	Remote Procedure Call	57470	mNW Info	1

								RPC Exploits			
15 Feb 2002 22:38:46	cjacobs	win32	Zone Alarm	64.194.x.x	1	6	111	Remote Procedure Call RPC Exploits	47758	mNW Info	1
15 Feb 2002 21:02:35	emanon	Perl	iptables	12.226.x.x	1	6	80	HTTP HTTP Probe	-1	mNW Info	2
14 Feb 2002 09:54:45	Kimmy	win32	Zone Alarm	24.148.x.x	1	6	80	HTTP HTTP Probe	63007	mNW Info	2
14 Feb 2002 03:20:15	mikem_nj	win32	Zone Alarm	68.37.x.x	1	6	80	HTTP HTTP Probe	49687	mNW Info	1
14 Feb 2002 03:00:12	Dr. Who	win32	Zone Alarm	68.37.x.x	1	6	80	HTTP HTTP Probe	37771	mNW Info	1
13 Feb 2002 23:18:35	tinyalien	win32	NetGear	68.37.x.x	1	6	80	HTTP HTTP Probe	42560	mNW Info	3
13 Feb 2002 00:57:12	timlu	Perl	ipchains	213.67.x.x	1	6	80	HTTP HTTP Probe	-1	mNW Info	2
12 Feb 2002 23:59:31	OmegaCop	win32	BlackICE	213.67.x.x	1	6	80	HTTP HTTP port probe	51839	advICE | mNW Info	1
12 Feb 2002 23:51:35	PsiCop	Perl	iptables	213.67.x.x	1	6	80	HTTP HTTP Probe	-1	mNW Info	2
12 Feb 2002 23:46:33	rrayge	win32	Dlink/SMC	24.8.x.x	1	6	111	Remote Procedure Call RPC Exploits	38337	mNW Info	2
12 Feb 2002 23:42:12	RipSpace	win32	Zone Alarm	216.222.x.x	1	6	111	Remote Procedure Call RPC Exploits	39002	mNW Info	1
12 Feb 2002 23:35:48	Mats	win32	Zone Alarm	213.67.x.x	1	6	80	HTTP HTTP Probe	59171	mNW Info	1
12 Feb 2002 23:24:58	wheel1	win32	Zone Alarm	213.67.x.x	1	6	80	HTTP HTTP Probe	36095	mNW Info	1
12 Feb 2002 20:36:49	Kukapa	win32	Zone Alarm	212.83.x.x	1	6	111	Remote Procedure Call RPC Exploits	49727	mNW Info	1
12 Feb 2002 11:35:44	Beampiper	win32	Zone Alarm	203.54.x.x	1	6	111	Remote Procedure Call RPC Exploits	64708	mNW Info	1

11 Feb 2002 23:47:50	suncatcher	win32	BlackICE	68.63.x.x	1	6	80	HTTP HTTP port probe	62454	advICE | mNW Info	1
11 Feb 2002 23:32:01	Merger	win32	BlackICE	68.63.x.x	1	6	80	HTTP HTTP port probe	52336	advICE | mNW Info	1
11 Feb 2002 20:03:15	Wombatz	Perl	SonicWall	198.144.x.x	1	6	111	Remote Procedure Call RPC Exploits	-1	mNW Info	2
11 Feb 2002 09:50:24	Lew	win32	Zone Alarm	68.45.x.x	1	6	80	HTTP HTTP Probe	41368	mNW Info	1
11 Feb 2002 03:11:48	eric.d	win32	BlackICE	24.95.x.x	1	6	111	Remote Procedure Call RPC port probe	50228	advICE | mNW Info	1
11 Feb 2002 01:33:12	wayoutthere	win32	Zone Alarm	206.126.x.x	1	6	111	Remote Procedure Call RPC Exploits	58976	mNW Info	2
10 Feb 2002 19:45:24	AirCoTek	win32	Zone Alarm	65.15.x.x	1	6	111	Remote Procedure Call RPC Exploits	40143	mNW Info	1
10 Feb 2002 17:48:57	TazMainiac	Perl	iptables	24.6.x.x	1	6	111	Remote Procedure Call RPC Exploits	-1	mNW Info	2
DATA	TRUNCATED	HERE									

List ALL Incident Activity

Activity Date (UTC): 16 Feb 2002 22:32:26

```
Standard escalation email sent to: security@public.hr.hl.cn
```

An interesting correlation comes from a log file running on the target host:

#Software: Microsoft Internet Information Services 5.0
#Version: 1.0
#Date: 2002-02-16 07:39:12
#Fields: date time c-ip cs-username s-ip s-port cs-method cs-uri-stem cs-uri-query sc-status cs(User-Agent)
2002-02-16 07:39:11 218.7.3.19 - MY.NET.0.1 80 GET /winnt/system32/cmd.exe /c+dir 404 -
2002-02-16 07:39:11 218.7.3.19 - MY.NET.0.1 80 GET /winnt/system32/cmd.exe /c+dir 404 -
2002-02-16 07:39:14 218.7.3.19 - MY.NET.0.1 80 GET /scripts/..Á%pc../winnt/system32/cmd.exe /c+dir 500 -
2002-02-16 07:39:14 218.7.3.19 - MY.NET.0.1 80 GET /scripts/..À%9v../winnt/system32/cmd.exe /c+dir 500 -

2002-02-16 07:39:15 218.7.3.19 - MY.NET.0.1 80 GET /scripts/..À%qf../winnt/system32/cmd.exe /c+dir 500 -
2002-02-16 07:39:15 218.7.3.19 - MY.NET.0.1 80 GET /scripts/..Á%8s../winnt/system32/cmd.exe /c+dir 500 -
2002-02-16 07:39:17 218.7.3.19 - MY.NET.0.1 80 GET /scripts/..Á ../winnt/system32/cmd.exe /c+dir 500 -
2002-02-16 07:39:17 218.7.3.19 - MY.NET.0.1 80 GET /winnt/system32/cmd.exe /c+dir 404 -
2002-02-16 07:39:18 218.7.3.19 - MY.NET.0.1 80 GET /scripts/..o../winnt/system32/cmd.exe /c+dir 404 -
2002-02-16 07:39:18 218.7.3.19 - MY.NET.0.1 80 GET /winnt/system32/cmd.exe /c+dir 404 -
2002-02-16 07:39:20 218.7.3.19 - MY.NET.0.1 80 GET /scripts/..ð€€¯../winnt/system32/cmd.exe /c+dir 404 -
2002-02-16 07:39:20 218.7.3.19 - MY.NET.0.1 80 GET /scripts/..ø€€€¯../winnt/system32/cmd.exe /c+dir 404 -
2002-02-16 07:39:22 218.7.3.19 - MY.NET.0.1 80 GET /scripts/..ü€€€€¯../winnt/system32/cmd.exe /c+dir 404 -
2002-02-16 07:39:23 218.7.3.19 - MY.NET.0.1 80 GET /winnt/system32/cmd.exe /c+dir 404 –

This behavior is consistent with the sadmind/IIS attack, but the time stamp in this log differs from that of the IDS sensor approximately 5 hours. This not being a server I administer, I'm not sure if the time on that server is incorrect, set to a different time zone, or has been modified by a successful attack. No other traces are present indicating a successful attack, so I would lean towards one of the other two causes of the difference. A review of the entire capture for any occurrence of the attacking IP address shows it only in the timeframe of the detect, and no activity later that morning as this might suggest. This location uses NAT, so the "MY.NET" address here is the internal address that is recorded in the server's log file. The last field of the log file shown here is the http status code of the request on that line. According to rfc2616 (http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html), http status code 500 means: "Internal Server Error - The server encountered an unexpected condition which prevented it from fulfilling the request." Based on that code, and the absence of traffic after the initial exploit attempt, it is most likely that this attack did not succeed.

CERT® Advisory CA-2001-11 sadmind/IIS Worm (http://www.cert.org/advisories/CA-2001-11.html) - describes the vulnerability and typical footprint of this attack, which is consistent with the observed behavior of this detect.

Evidence of Active Targeting:

Moderate to high - After discovering this attack, a review of the portscan.log for this IP address reveals:

Feb 15 23:23:45 218.7.3.19:35765 -> MY.NET.CLASSC.169:80 SYN ******S*
Feb 15 23:23:45 218.7.3.19:35766 -> MY.NET.CLASSC.170:80 SYN ******S*
Feb 15 23:23:45 218.7.3.19:35767 -> MY.NET.CLASSC.171:80 SYN ******S*
Feb 15 23:23:45 218.7.3.19:35768 -> MY.NET.CLASSC.172:80 SYN ******S*
Feb 15 23:23:45 218.7.3.19:35769 -> MY.NET.CLASSC.173:80 SYN ******S*

Note that the timestamp Snort logged this scan as is: Feb 15 23:23:45 , just several hours prior to the attack seen in this detect. This would lead me to believe that this attacker discovered the IIS server during what was most likely a non-targeted broad host scan looking for targets of opportunity, and returned to actively exploit this particular IP address later. Other than the portscan, MY.NET.CLASSC.170 is the only IP attacked by this host..

Severity:

Criticality – This server is the core mail server.
Criticality = 4

Lethality – The server is running IIS to support Outlook Web Access, and this is an IIS exploit. Had this succeeded the attacked could have run code as the IUSR_machinename user.
Lethality = 3.

System Countermeasures – The system has all patches applied for known & patchable vulnerabilities.
System Countermeasures = 5.

Network Countermeasures – Site has a managed firewall, so most likely this scan was stopped at the firewall. A firewall rule audit has not been completed, so lowering this score 1 point. Assuming the managed security provider's rules are appropriate is dangerous, and regular audits ensure that rules are up to date and applicable to the current needs of the company. Apart from my traffic analysis as part of this practical, no intrusion detection systems are in use.

Network Countermeasures = 3

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

(4 + 3) – (5 + 3) = -1

Defensive Recommendation:

1. Consider using a VPN to allow any remote access to this server and block external port 80 directly to this box.
2. Implement a method to synchronize the time across all network devices and servers. While this does not affect this attack, the lack of synchronized time makes a legal case difficult to prosecute, which could leave you unable to avail yourself of a legal remedy if you are successfully attacked.

3. Ensure all patches are kept up to date on this machine, and review the log files regularly for unexpected entries. Specifically ensure that the following patches are applied:
    a. Microsoft Bulletin MS00-78 addresses the Folder Traversal vulnerability. http://www.microsoft.com/technet/security/bulletin/MS00-078.asp

    b. Microsoft Bulletin MS00-057 for the *File Permission Canonicalization* vulnerability has a patch that also protects against the *Web Server Folder Traversal* vulnerability. http://www.microsoft.com/technet/security/bulletin/ms00-057.asp
4. Implement SSL with 128 bit encryption for Outlook Web Access and refuse all port 80 traffic. The 128 bit encryption by itself does not improve your defensive posture for this detect, but will improve your overall defenses in this area.

Multiple Choice Test Question:

What would have made this detect have a higher lethality?
a. If the target machine was a DNS server.
b. If the target machine was a linux server
c. If the exploit could obtain super-user access
d. If the target machine was the firewall

Answer = c

Network Detect #2 - Hack'a'Tack Trojan port scan

```
Feb 12 03:30:35 24.161.199.95:31790 -> MY.NET.CLASSC.169:31789 UDP
Feb 12 03:30:35 24.161.199.95:31790 -> MY.NET.CLASSC.171:31789 UDP
Feb 12 03:30:35 24.161.199.95:31790 -> MY.NET.CLASSC.173:31789 UDP
Feb 12 03:30:35 24.161.199.95:31790 -> MY.NET.CLASSC.170:31789 UDP
Feb 12 03:30:35 24.161.199.95:31790 -> MY.NET.CLASSC.172:31789 UDP
```

Source of Trace:

The network of one of my customers.

Detect Generated By:

Snort v 1.83

Probability the Source Address was Spoofed:

Low –The response has to be received from the trojan in order for the person scanning to known they have found a compromised machine. While there are possibilities that someone could be listening between this address and the spoofed address, that is a low probability.

Description of the Attack:

This was a UDP port scan with a source port of 31790 and a destination port of 31789. This is typical behavior for the for the Hack'a'Tack trojan.

My rules did not include the following current snort rule which would have alerted on this traffic:
"alert tcp $EXTERNAL_NET 31790 -> $HOME_NET 31789 (msg:"SCAN trojan hack-a-tack probe"; content: "A"; depth: 1; reference:arachnids,314; flags:A+; classtype:attempted-recon; sid:614; rev:1;)" (http://www.snort.org/snort-db/sid.html?id=614)

I found this attack reviewing the portscan.log looking for patterns of port scans that I had not seen before. When I saw this pattern, I dug further using tcpdump and looked for any packets with both of those ports. I found the this IP address had several probes in addition to the one listed above:

```
11:26:14.304903 24.161.199.95.31790 > MY.NET.CLASSC.169.31789:  udp 1
11:26:14.312422 24.161.199.95.31790 > MY.NET.CLASSC.170.31789:  udp 1
11:26:14.313009 24.161.199.95.31790 > MY.NET.CLASSC.171.31789:  udp 1
11:26:14.313664 24.161.199.95.31790 > MY.NET.CLASSC.173.31789:  udp 1
11:26:14.321525 24.161.199.95.31790 > MY.NET.CLASSC.172.31789:  udp 1
17:25:50.444703 24.161.199.95.31790 > MY.NET.CLASSC.169.31789:  udp 1
17:25:50.446414 24.161.199.95.31790 > MY.NET.CLASSC.171.31789:  udp 1
17:25:50.446922 24.161.199.95.31790 > MY.NET.CLASSC.173.31789:  udp 1
17:25:50.455232 24.161.199.95.31790 > MY.NET.CLASSC.170.31789:  udp 1
17:25:50.455786 24.161.199.95.31790 > MY.NET.CLASSC.172.31789:  udp 1
19:14:34.836854 24.161.199.95.31790 > MY.NET.CLASSC.169.31789:  udp 1
19:14:34.848344 24.161.199.95.31790 > MY.NET.CLASSC.170.31789:  udp 1
19:14:34.854317 24.161.199.95.31790 > MY.NET.CLASSC.171.31789:  udp 1
```

19:14:34.857341 24.161.199.95.31790 > MY.NET.CLASSC.173.31789: udp 1
19:14:34.862594 24.161.199.95.31790 > MY.NET.CLASSC.172.31789: udp 1
00:00:52.218541 24.161.199.95.31790 > MY.NET.CLASSC.169.31789: udp 1
00:00:52.218755 24.161.199.95.31790 > MY.NET.CLASSC.171.31789: udp 1
00:00:52.219844 24.161.199.95.31790 > MY.NET.CLASSC.173.31789: udp 1
00:00:52.226688 24.161.199.95.31790 > MY.NET.CLASSC.170.31789: udp 1
00:00:52.227336 24.161.199.95.31790 > MY.NET.CLASSC.172.31789: udp 1
04:38:06.719860 24.161.199.95.31790 > MY.NET.CLASSC.169.31789: udp 1
04:38:06.726294 24.161.199.95.31790 > MY.NET.CLASSC.171.31789: udp 1
04:38:06.728817 24.161.199.95.31790 > MY.NET.CLASSC.173.31789: udp 1
04:38:06.729892 24.161.199.95.31790 > MY.NET.CLASSC.170.31789: udp 1
04:38:06.732864 24.161.199.95.31790 > MY.NET.CLASSC.172.31789: udp 1
06:16:13.403676 24.161.199.95.31790 > MY.NET.CLASSC.169.31789: udp 1
06:16:13.404189 24.161.199.95.31790 > MY.NET.CLASSC.171.31789: udp 1
06:16:13.410427 24.161.199.95.31790 > MY.NET.CLASSC.170.31789: udp 1
06:16:13.416090 24.161.199.95.31790 > MY.NET.CLASSC.173.31789: udp 1
06:16:13.420698 24.161.199.95.31790 > MY.NET.CLASSC.172.31789: udp 1
03:30:35.949887 24.161.199.95.31790 > MY.NET.CLASSC.169.31789: udp 1
03:30:35.955647 24.161.199.95.31790 > MY.NET.CLASSC.171.31789: udp 1
03:30:35.956482 24.161.199.95.31790 > MY.NET.CLASSC.173.31789: udp 1
03:30:35.956652 24.161.199.95.31790 > MY.NET.CLASSC.170.31789: udp 1
03:30:35.961549 24.161.199.95.31790 > MY.NET.CLASSC.172.31789: udp 1
04:19:05.574044 24.161.199.95.31790 > MY.NET.CLASSC.169.31789: udp 1
04:19:05.575439 24.161.199.95.31790 > MY.NET.CLASSC.171.31789: udp 1
04:19:05.581499 24.161.199.95.31790 > MY.NET.CLASSC.173.31789: udp 1
04:19:05.584014 24.161.199.95.31790 > MY.NET.CLASSC.170.31789: udp 1
04:19:05.587128 24.161.199.95.31790 > MY.NET.CLASSC.172.31789: udp 1
05:04:57.774987 24.161.199.95.31790 > MY.NET.CLASSC.169.31789: udp 1
05:04:57.776596 24.161.199.95.31790 > MY.NET.CLASSC.171.31789: udp 1
05:04:57.777123 24.161.199.95.31790 > MY.NET.CLASSC.173.31789: udp 1
05:04:57.782709 24.161.199.95.31790 > MY.NET.CLASSC.172.31789: udp 1
17:16:02.170018 24.161.199.95.31790 > MY.NET.CLASSC.169.31789: udp 1
17:16:02.170614 24.161.199.95.31790 > MY.NET.CLASSC.171.31789: udp 1
17:16:02.172359 24.161.199.95.31790 > MY.NET.CLASSC.173.31789: udp 1
17:16:02.178018 24.161.199.95.31790 > MY.NET.CLASSC.172.31789: udp 1
17:16:02.178614 24.161.199.95.31790 > MY.NET.CLASSC.170.31789: udp 1

Since none of the company's systems responded, either the firewall blocked the inbound traffic, or there are no Trojans on internal systems to answer. I do not have access to the firewall logs, which would be a key point of correlation to be able to make that assessment without a very broad assumption based solely on the network traffic.

CVE Reference: CAN-1999-0660 (under review)

Attack Mechanism:

This trojan is often installed when a user executes the file "server.exe". The file is most commonly sent to a user via email or instant messaging programs, disguised as something else in order to entice the user to execute it. The execution places the file Expl32.exe on their system, and adds it to the "HKLM\Software\Microsoft\Windows\CurrentVersion\Run" registry key. As a result, each time the machine is booted, the trojan starts and listens on tcp port 31785 or UDP port 31789 or 31791.

From Internet Security Systems Security Alert #30, "Windows Backdoor Update III" at http://xforce.iss.net/alerts/advise30.php:

> "Hack'a'Tack is a backdoor that allows attackers to move and kill windows on your desktop, open an FTP server on your machine, log keystrokes, save passwords you type, shut down the machine, and upload, download, and execute files. Hack'a'Tack only runs on Windows 95 and 98. Hack'a'Tack uses TCP port 31785 and UDP ports 31789 and 31791. If you connect to TCP port 31785, it will display a banner such as: hostxforce.org (In this example, xforce.org is the hostname of the machine) If you see TCP port 31785 and UDP ports 31789 and 31791 open when you run 'netstat -a', then you probably have Hack'a'Tack on your machine."

Correlations:

Dshield.org: http://www.dshield.org/ipinfo.php?ip=24.161.199.95
Shows 1592 records of this IP address against 1152 different addresses.

My NetWatchman:
The correlation from MyNetWatchman.com shows this IP address running various scans across a number of IP address ranges for port 31789. Most of the
The full incident log from MyNetWatchman can be accessed via : http://www.mynetwatchman.com/LID.asp?IID=2796908

Incident Id : 2796908	Source Ip : 24.161.199.95

Source Name :	Provider Domain : rr.com
Net Bios Name :	DNS Name : bak-24-161-199-95.bak.rr.com
Total Event Count : 123	Total Distinct Agent : 32/11200
Response : No Recent Activity	
Status Description : Closed	
Exclusion Reason :	

Network Name/NextNIC	Start IP - End IP
ROAD-RUNNER-5/DUMMY	24.160.0.0 - 24.170.127.255

```
NextNIC:99999
Whois provider: rr.com
```

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

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

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

   Domain System inverse mapping provided by:

   DNS1.RR.COM                24.30.200.3
   DNS2.RR.COM                24.30.201.3

   Record last updated on 11-Jul-2000.
   Database last updated on 23-May-2001 22:44:44 EDT.

The ARIN Registration Services Host contains ONLY Internet
Network Information: Networks, ASN's, and related POC's.
Please use the whois server at rs.internic.net for DOMAIN related
```
Information and whois.nic.mil for NIPRNET Information.

Most Recent Event	Agent Alias	Agent Type	Log Type	Target Ip	# of IPs	IP	Target	Port/	Source	Explanation	Event

Date/Time (UTC)					Targeted	Protocol	Port	Issue Description	Port		Count
14 Feb 2002 22:37:33	jabach1v	win32	Zone Alarm	66.130.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	6
14 Feb 2002 21:49:34	Lee	win32	Zone Alarm	66.140.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	2
14 Feb 2002 21:30:08	Two_Cycle	win32	Zone Alarm	66.133.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	11
14 Feb 2002 19:39:22	chadvavra	win32	Zone Alarm	192.168.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	6
14 Feb 2002 12:06:18	NetNark	win32	Zone Alarm	66.188.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	4
14 Feb 2002 11:51:11	Bravo_19	win32	Zone Alarm	66.183.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	5
14 Feb 2002 11:30:57	davidol	win32	Zone Alarm	66.176.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	5
14 Feb 2002 10:31:52	jmain6	win32	Zone Alarm	66.156.x.x	2	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	6
14 Feb 2002 10:31:22	Wilson Phillips	win32	BlackICE	192.168.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	7
14 Feb 2002 09:34:23	NOT ME	win32	Zone Alarm	66.137.x.x	3	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	4
14 Feb 2002 06:26:03	Hackhater	win32	BlackICE	66.125.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	5
13 Feb 2002 12:22:03	jroos	win32	Linksys	66.188.x.x	1	6	31789	Unassigned Hack'a'Tack Probe	31790	mNW Info	3
13 Feb 2002 10:58:06	RenL	win32	BlackICE	66.169.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	2
13 Feb 2002 09:58:26	jm_wells	win32	BlackICE	66.156.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	1
12 Feb 2002 10:43:44	seymourphoto	win32	Zone Alarm	66.177.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	2
11 Feb 2002 05:02:16	Bester	win32	BlackICE	66.92.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	5
11 Feb 2002 05:00:25	tsh	Perl	ipchains	66.92.x.x	5	17	31789	Unassigned Hack'a'Tack	-1	mNW Info	20

10 Feb 2002 19:57:42	Mindy	win32	Zone Alarm	66.92.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	3
10 Feb 2002 19:53:26	jonivan45	win32	Zone Alarm	66.92.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	3
10 Feb 2002 14:21:12	algae	win32	BlackICE	66.188.x.x	1	17	31789	Unassigned Hack'a'Tack	-1	mNW Info	1
10 Feb 2002 12:00:53	ADSLMike	win32	Zone Alarm	66.157.x.x	2	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	4
10 Feb 2002 11:55:22	dr_vms	win32	BlackICE	66.156.x.x	2	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	2
10 Feb 2002 07:06:33	Volunteer	win32	BlackICE	66.92.x.x	2	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	6
9 Feb 2002 12:10:16	Shadowmaker	win32	Zone Alarm	66.183.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	1
9 Feb 2002 10:42:12	gtpryor	win32	BlackICE	10.138.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	1
9 Feb 2002 03:16:28	CiscoKid	win32	Zone Alarm	66.123.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	1
8 Feb 2002 23:09:47	peyoteman	win32	Zone Alarm	66.157.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	1
8 Feb 2002 23:06:54	honeytw	win32	BlackICE	66.157.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	1
8 Feb 2002 10:33:56	jcarone	win32	Dlink/SMC	66.87.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	1
8 Feb 2002 10:10:44	IcePrick	win32	BlackICE	10.90.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	1
8 Feb 2002 06:55:28	RedOregon	win32	Zone Alarm	24.162.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	2
8 Feb 2002 05:55:52	scooby	win32	Zone Alarm	24.171.x.x	1	17	31789	Unassigned Hack'a'Tack	31790	mNW Info	1

List ALL Incident Activity

Activity Date (UTC): 14 Feb 2002 19:02:11

```
Standard escalation email sent to: security@rr.com
```

Evidence of Active Targeting:

Based on the correlations with the number of other IP address ranges being scanned there is a very low probability that this is targeted to the company. Consistent with the type of attack, this scan is looking for Trojans, regardless of where they may be.

Severity:

Criticality – This scan ran against all the public IP addresses of the company, however it only affects Windows 9x systems. None of the core systems are Windows 9x systems, however confidential company information could be revealed if a system was compromised.
Criticality = 3

Lethality – In order for this scan to produce results, the scan has to be allowed through the firewall (I'm assuming it was stopped, though I would recommend a Firewall log audit to confirm this) and hit a machine with the trojan.
Lethality = 2.

System Countermeasures – All systems have updated anti-virus signatures.
System Countermeasures = 5.

Network Countermeasures – Site has a managed firewall, so most likely this scan was stopped at the firewall. A firewall rule audit has not been completed, so lowering this score 1 point. Assuming the managed security provider's rules are appropriate is dangerous, and regular audits ensure that rules are up to date and applicable to the current needs of the company. Apart from my traffic analysis as part of this practical, no intrusion detection systems are in use.
Network Countermeasures = 3

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

(3 + 2) – (5 + 3) = -3

Defensive Recommendation:

1. Automate anti-virus signature updates to ensure as up to date as possible.
2. Block attachments to email named "server.exe" . While this does not prevent a renamed file from coming through, every little bit helps!

3. Educate employees regularly on social engineering attacks, and about the dangers of executing attachments from non-expected sources.
4. Establish a knowledgeable "go-to" person in the office to query when employees are unsure of an attachment from email. This person should receive more in depth training on suspicious activity, and what to do when they encounter it.

Multiple Choice Test Question:

What tcpdump filter would show only typical hack-a-tack probe traffic?

A. tcpdump '(tcp or ip) and (dst port 31789)'
B. tcpdump 'src port 31789 and dst port 31790'
C. tcpdump `src port 31790 and dst port 31789'
D. tcpdump 'src port 31790 or dst port 31790'

Answer = C

Network Detect #3 - Code Red v2

[**] [1:1256:3] WEB-IIS CodeRed v2 root.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
02/16-03:25:26.647724 66.76.77.48:4832 -> MY.NET.CLASSC.170:80
TCP TTL:115 TOS:0x0 ID:26092 IpLen:20 DgmLen:112 DF
AP Seq: 0xE74AC174 Ack: 0x4A529D53 Win: 0x4470 TcpLen: 20
[Xref => http://www.cert.org/advisories/CA-2001-19.html]

[**] [1:1256:3] WEB-IIS CodeRed v2 root.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
02/16-03:25:26.843748 66.76.77.48:4832 -> MY.NET.CLASSC.170:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:112
AP Seq: 0x74C14AE7 Ack: 0x74C14AE7 Win: 0x4428 TcpLen: 20
[Xref => http://www.cert.org/advisories/CA-2001-19.html]

[**] [1:1256:3] WEB-IIS CodeRed v2 root.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
02/16-03:25:27.137076 66.76.77.48:4889 -> MY.NET.CLASSC.170:80
TCP TTL:115 TOS:0x0 ID:26172 IpLen:20 DgmLen:162 DF
AP Seq: 0xE75FC315 Ack: 0x4A558268 Win: 0x4470 TcpLen: 20
[Xref => http://www.cert.org/advisories/CA-2001-19.html]

An interesting note: If you observe the TTL values for all three of the alerts, the 1st and 3rd packets have a TTL of 115, which is reasonable considering this attack originates from Windows hosts, and often the starting TTL value for Windows hosts is 128. The 2nd packet however has a TTL of 255, which is inconsistent with the other two packets. In addition the sequence numbers are not in order as expected (unless packets arrived out of order and then they would still be close to each other), and further caused me to question the 2nd alert.

"Packet #1" Snort Timestamp - 02/16-03:25:26.647724 , Seq # 0xE74AC174
"Packet #2" Snort Timestamp - 02/16-03:25:26.843748 , Seq # 0x74C14AE7
"Packet #3" Snort Timestamp - 02/16-03:25:27.137076 , Seq # 0xE75FC315

"Packet #2" appears to be an ack from my server back to the attacker, so I'm puzzled about why the alert fired on that packet and reported it as 66.76.77.48:4832 -> MY.NET.CLASSC.170:80 and the TTL for that packet in the capture is 127, not 255. This does not appear in the raw traffic, so most likely it's a problem with Snort. I need to gather more information and research this anomaly further.

Summary of traffic between the two hosts on port 4832 and port 80(the first two alerts).

=+

02/16-03:25:26.575601 66.76.77.48:4832 -> MY.NET.CLASSC.170:80
TCP TTL:115 TOS:0x0 ID:26085 IpLen:20 DgmLen:48 DF
******S* Seq: 0xE74AC173 Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

=+

02/16-03:25:26.577224 MY.NET.CLASSC.170:80 -> 66.76.77.48:4832
TCP TTL:127 TOS:0x0 ID:49656 IpLen:20 DgmLen:48 DF

```
***A**S* Seq: 0x4A529D52  Ack: 0xE74AC174  Win: 0x4470  TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

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

02/16-03:25:26.636690 66.76.77.48:4832 -> MY.NET.CLASSC.170:80
TCP TTL:115 TOS:0x0 ID:26091 IpLen:20 DgmLen:40 DF
***A**** Seq: 0xE74AC174  Ack: 0x4A529D53  Win: 0x4470  TcpLen: 20

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

02/16-03:25:26.647724 66.76.77.48:4832 -> MY.NET.CLASSC.170:80
TCP TTL:115 TOS:0x0 ID:26092 IpLen:20 DgmLen:112 DF
***AP*** Seq: 0xE74AC174  Ack: 0x4A529D53  Win: 0x4470  TcpLen: 20
47 45 54 20 2F 73 63 72 69 70 74 73 2F 72 6F 6F  GET /scripts/roo
74 2E 65 78 65 3F 2F 63 2B 64 69 72 20 48 54 54  t.exe?/c+dir HTT
50 2F 31 2E 30 0D 0A 48 6F 73 74 3A 20 77 77 77  P/1.0..Host: www
0D 0A 43 6F 6E 6E 6E 65 63 74 69 6F 6E 3A 20 63  ..Connnection: c
6C 6F 73 65 0D 0A 0D 0A                          lose....

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

02/16-03:25:26.843748 MY.NET.CLASSC.170:80 -> 66.76.77.48:4832
TCP TTL:127 TOS:0x0 ID:49657 IpLen:20 DgmLen:40 DF
***A**** Seq: 0x4A529D53  Ack: 0xE74AC1BC  Win: 0x4428  TcpLen: 20

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

02/16-03:25:26.980743 MY.NET.CLASSC.170:80 -> 66.76.77.48:4832
TCP TTL:127 TOS:0x0 ID:49658 IpLen:20 DgmLen:228 DF
***AP*** Seq: 0x4A529D53  Ack: 0xE74AC1BC  Win: 0x4428  TcpLen: 20
48 54 54 50 2F 31 2E 31 20 32 30 30 20 4F 4B 0D  HTTP/1.1 200 OK.
0A 53 65 72 76 65 72 3A 20 4D 69 63 72 6F 73 6F  .Server: Microso
66 74 2D 49 49 53 2F 35 2E 30 0D 0A 44 61 74 65  ft-IIS/5.0..Date
3A 20 53 61 74 2C 20 31 36 20 46 65 62 20 32 30  : Sat, 16 Feb 20
30 32 20 30 38 3A 32 33 3A 32 39 20 47 4D 54 0D  02 08:23:29 GMT.
```

```
0A 43 6F 6E 74 65 6E 74 2D 54 79 70 65 3A 20 61  .Content-Type: a

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

02/16-03:25:26.983042 MY.NET.CLASSC.170:80 -> 66.76.77.48:4832
TCP TTL:127 TOS:0x0 ID:49659 IpLen:20 DgmLen:1500 DF
***AP*** Seq: 0x4A529E0F  Ack: 0xE74AC1BC  Win: 0x4428  TcpLen: 20
20 44 69 72 65 63 74 6F 72 79 20 6F 66 20 63 3A   Directory of c:
5C 69 6E 65 74 70 75 62 5C 73 63 72 69 70 74 73  \inetpub\scripts
0D 0A 0D 0A 30 32 2F 31 30 2F 32 30 30 32 20 20  ....02/10/2002
30 33 3A 30 39 61 20 20 20 20 20 20 3C 44 49 52  03:09a      <DIR
3E 20 20 20 20 20 20 20 20 20 2E 0D 0A 30 32  >          ...02
2F 31 30 2F 32 30 30 32 20 20 30 33 3A 30 39 61  /10/2002  03:09a

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

02/16-03:25:26.983407 MY.NET.CLASSC.170:80 -> 66.76.77.48:4832
TCP TTL:127 TOS:0x0 ID:49660 IpLen:20 DgmLen:420 DF
***AP*** Seq: 0x4A52A3C3  Ack: 0xE74AC1BC  Win: 0x4428  TcpLen: 20
30 39 3A 33 32 70 20 20 20 20 20 20 20 20 20 20  09:32p
20 20 20 20 20 20 20 20 20 30 20 54 46 54 50 31           0 TFTP1
37 39 36 0D 0A 30 39 2F 31 39 2F 32 30 30 31 20  796..09/19/2001
20 30 32 3A 31 36 61 20 20 20 20 20 20 20 20 20   02:16a
20 20 20 20 20 20 20 20 20 20 30 20 54 46 54 50           0 TFTP
31 38 36 38 0D 0A 31 30 2F 32 39 2F 32 30 30 31  1868..10/29/2001

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

02/16-03:25:27.059130 66.76.77.48:4832 -> MY.NET.CLASSC.170:80
TCP TTL:115 TOS:0x0 ID:26157 IpLen:20 DgmLen:40 DF
*****R** Seq: 0xE74AC1BC  Ack: 0xE759BEB6  Win: 0x0  TcpLen: 20

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

02/16-03:25:27.079110 66.76.77.48:4832 -> MY.NET.CLASSC.170:80
TCP TTL:115 TOS:0x0 ID:26162 IpLen:20 DgmLen:40
```

```
*****R** Seq: 0xE74AC1BC  Ack: 0xE74AC1BC  Win: 0x0  TcpLen: 20

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

02/16-03:25:27.080710 66.76.77.48:4832 -> MY.NET.CLASSC.170:80
TCP TTL:115 TOS:0x0 ID:26163 IpLen:20 DgmLen:40
*****R** Seq: 0xE74AC1BC  Ack: 0xE74AC1BC  Win: 0x0  TcpLen: 20

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

Source of Trace:

My customer's network.

Detect Generated By:

Snort v 1.83

Rule that generated this alert:

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-IIS CodeRed v2 root.exe access"; flags: A+; uricontent:"scripts/root.exe?"; nocase; classtype:web-application-attack; reference:url,www.cert.org/advisories/CA-2001-19.html; sid:1256; rev:3;)
```

Probability the Source Address was Spoofed:

Low, since the Code Red worm has known behavior of attempting to propagate itself once it has compromised a host, there is no need to spoof since the source host in no way points back to the person that created and launched the original attack. There is something of note in the

Description of the Attack:

The Code Red v2 worm is a self propagating worm that from the 1st -19th of the month attempts to infect as many systems as possible, and on the 20th day of the month will try to do a denial of service against www.whitehouse.gov. Day 21 until the end of the month it lies dormant, until it wakes up on the 1st to repeat the cycle. This variant will place a trojan on the system, creating a back door for access later. Some variants will deface the local web page with "http://www.worm.com!, Hacked By Chinese!". I referenced eEye's analysis (http://www.eeye.com/html/Research/Advisories/AL20010717.html), and the F-Secure's Code Red description (http://www.europe.f-secure.com/v-descs/bady.shtml). Both are excellent resources for understanding this worm.

Attack Mechanism:

The attack mechanism is a scan for machines that are vulnerable to the Microsoft Indexing Service (.ida) remote buffer overflow, and plant the worm using that vulnerability. Once the worm is in place, it plants a Trojan, modifies the registry, then it spawns multiple threads (100 on English systems) to attempt to infect other systems and sometimes deface a web page local to the system.

Correlations:

Dshield.org: http://www.dshield.org/ipinfo.php?ip=66.76.77.48
Shows 846 records of this IP address against 266 different addresses.

My NetWatchman:
The correlation from MyNetWatchman.com shows this IP address running various scans across a number of IP address ranges for port 80. The full incident log from MyNetWatchman can be accessed via :
http://www.mynetwatchman.com/LID.asp?IID=2847227

Incident Id : 2847227	Source Ip : 66.76.77.48
Source Name :	Provider Domain : cox-internet.com
Net Bios Name :	DNS Name : cdm-66-77-48-mwel.cox-internet.com
Total Event Count : 156	Total Distinct Agent : 14/5750
Response : No Recent Activity	
Status Description : Closed	
Exclusion Reason :	

Network Name/NextNIC	Start IP - End IP
TCAC-2/DUMMY	66.76.0.0 - 66.76.255.255
TCAC-2/DUMMY	66.76.0.0 - 66.76.191.255
TCAC-2/DUMMY	66.76.0.0 - 66.76.127.255

```
NextNIC:99999
Whois provider: tca.net
```

```
TCA Internet (NETBLK-TCAC-2)
   3314 SSW Loop 323
   Tyler, TX 75701
   US

   Netname: TCAC-2
   Netblock: 66.76.0.0 - 66.76.127.255
   Maintainer: TCAC

   Coordinator:
      Strout, Jeff  (JS2407-ARIN)  jeff.strout@cox.com
      903-939-7200

   Domain System inverse mapping provided by:

   ROSE.TYLER.NET              205.218.118.1
   NS.TCA.NET                  208.180.0.2

   ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE

   Record last updated on 20-Apr-2001.
   Database last updated on 19-Jul-2001 23:08:10 EDT.

The ARIN Registration Services Host contains ONLY Internet
Network Information: Networks, ASN's, and related POC's.
Please use the whois server at rs.internic.net for DOMAIN related
```
Information and whois.nic.mil for NIPRNET Information.

Most Recent Event Date/Time (UTC)	Agent Alias	Agent Type	Log Type	Target Ip	# of IPs Targeted	IP Protocol	Target Port	Port/ Issue Description	Source Port	Explanation	Event Count
18 Feb 2002 09:26:50	RenL	win32	BlackICE	66.169.x.x	40	6	80	HTTP HTTP port probe	3864	advICE | mNW Info	75

18 Feb 2002 09:26:47	RenL	win32	BlackICE	66.34.x.x	4	6	80	HTTP IIS system32 command	-1	advICE | mNW Info	5
17 Feb 2002 00:51:48	jankemi	Perl	Cisco PIX	134.29.x.x	6	6	80	HTTP HTTP Probe	4564	mNW Info	15
17 Feb 2002 00:26:30	jonivan45	win32	Zone Alarm	66.92.x.x	1	6	80	HTTP HTTP Probe	1411	mNW Info	1
17 Feb 2002 00:07:39	tsh	Perl	ipchains	66.92.x.x	2	6	80	HTTP HTTP Probe	-1	mNW Info	6
16 Feb 2002 23:47:58	3	win32	Linksys	66.76.x.x	1	6	80	HTTP HTTP Probe	1271	mNW Info	13
16 Feb 2002 11:05:14	yellowbeard	win32	Dlink/SMC	66.1.x.x	1	6	80	HTTP HTTP Probe	4192	mNW Info	4
15 Feb 2002 05:34:55	chadvavra	win32	Zone Alarm	192.168.x.x	1	6	80	HTTP HTTP Probe	4635	mNW Info	1
14 Feb 2002 19:25:28	MSI	win32	Zone Alarm	66.157.x.x	1	6	80	HTTP HTTP Probe	3263	mNW Info	1
14 Feb 2002 17:35:26	davidol	win32	Zone Alarm	66.176.x.x	1	6	80	HTTP HTTP Probe	3946	mNW Info	1
14 Feb 2002 08:31:49	jm_wells	win32	BlackICE	66.156.x.x	1	6	80	HTTP HTTP port probe	4833	advICE | mNW Info	1
14 Feb 2002 06:14:12	Volunteer	win32	BlackICE	66.92.x.x	1	6	80	HTTP HTTP attack	-1	advICE | mNW Info	2
14 Feb 2002 06:14:12	Volunteer	win32	BlackICE	66.92.x.x	1	6	80	HTTP Suspicious URL	-1	advICE | mNW Info	26
14 Feb 2002 02:10:19	Lee	win32	Zone Alarm	66.140.x.x	1	6	80	HTTP HTTP Probe	1582	mNW Info	1
13 Feb 2002 20:51:36	donchicago	win32	Zone Alarm	66.1.x.x	1	6	80	HTTP HTTP Probe	1639	mNW Info	2
13 Feb 2002 20:17:26	pjwinpa	win32	NetGear	66.21.x.x	1	6	80	HTTP HTTP Probe	3567	mNW Info	2

List ALL Incident Activity

Activity Date (UTC): 17 Feb 2002 10:16:49

```
Standard escalation email sent to: abuse@cox-internet.com
```

CERT® Advisory CA-2001-13 Buffer Overflow In IIS Indexing Service DLL - http://www.cert.org/advisories/CA-2001-13.html

CVE - CAN-2001-0500 (under review) - http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2001-0500

F-Secure Code Red Description - http://www.europe.f-secure.com/v-descs/bady.shtml

Evidence of Active Targeting:

Given the nature of the worm and the observed scans from this IP address it is highly unlikely that this was a targeted attack.

Severity:

Criticality – This server is the core mail server.
Criticality = 4

Lethality – The server is running IIS to support Outlook Web Access, and this is an IIS exploit. Had this succeeded the resulting worm could have negatively impacted network traffic and disrupted business functions, not to mention require the mail server be taken offline to recover from the damage and verify integrity, since this variant of Code Red places a Trojan on the compromised machine.
Lethality = 4.

System Countermeasures – The system has all patches applied for known & patchable vulnerabilities.
System Countermeasures = 5.

Network Countermeasures – Site has a managed firewall, so most likely this scan was stopped at the firewall. A firewall rule audit has not been completed, so lowering this score 1 point. Assuming the managed security provider's rules are appropriate is dangerous, and regular audits ensure that rules are up to date and applicable to the current needs of the company. Apart from my traffic analysis as part of this practical, no intrusion detection systems are in use.

Network Countermeasures = 3

(Criticality + Lethality) –

(System Countermeasures + Network Countermeasures) = Severity

(4 + 4) – (5 + 3) = 0

Defensive Recommendation:

1. Ensure all patches are kept up to date on this machine, and review the log files regularly for unexpected entries. Specifically review the following bulletins and apply the associated patches:

   Microsoft Security Bulletin MS01-033 – "Unchecked Buffer in Index Server ISAPI Extension Could Enable Web Server Compromise"
   http://www.microsoft.com/technet/security/bulletin/ms01-033.asp

   Microsoft Security Bulletin MS00-052 – "'Relative Shell Path Vulnerability"
   http://www.microsoft.com/technet/security/bulletin/MS00-052.asp

2. Evaluate if this server really needs to be exposed to port 80 inbound from outside the network, consider alternate methods to provide remote mail services to employees.
3. Implement SSL with 128 bit encryption for Outlook Web Access and refuse all port 80 traffic. The 128 bit encryption by itself does not improve your defensive posture for this detect, but will improve your overall defenses in this area.

Multiple Choice Test Question:

What snort command line would display the packets as they are displayed in this detect, using "capture.dmp" as the source file?

A. snort -dvr capture.dmp '(host 66.76.77.48 and host MY.NET.CLASSC.170) and (port 80 and port 4832)'
B. snort -dvr capture.dmp '(host 66.76.77.48 and host MY.NET.CLASSC.170) and (dst port 80 or src port 4832)'
C. snort -dvr capture.dmp '(host 66.76.77.48 and host MY.NET.CLASSC.170) and (dst port 80 and src port 4832)'
D. snort -dvr '(host 66.76.77.48 and host MY.NET.CLASSC.170) and (port 80 or port 4832)' capture.dmp

Answer = A

Network Detect #4 - IIS ISAPI OVERFLOW IDA

```
[**] [1:1242:2] WEB-IIS ISAPI .ida access [**]
[Classification: access to a potentually vulnerable web application] [Priority: 2]
02/18-05:42:50.554025 140.125.31.117:4462 -> MY.NET.CLASSC.173:80
TCP TTL:107 TOS:0x0 ID:34973 IpLen:20 DgmLen:1500 DF
***AP*** Seq: 0xDE74A661  Ack: 0x25164FE1  Win: 0x4470  TcpLen: 20
[Xref => http://www.whitehats.com/info/IDS552]
[Xref => http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2000-0071]
```

The packet that caused the alert to fire:

```
02/18-05:42:50.554025 140.125.31.117:4462 -> MY.NET.CLASSC.173:80
TCP TTL:107 TOS:0x0 ID:34973 IpLen:20 DgmLen:1500 DF
***AP*** Seq: 0xDE74A661  Ack: 0x25164FE1  Win: 0x4470  TcpLen: 20
2F 64 65 66 61 75 6C 74 2E 69 64 61 3F 4E 4E 4E  /default.ida?NNN
4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E  NNNNNNNNNNNNNNNN
4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E  NNNNNNNNNNNNNNNN
4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E  NNNNNNNNNNNNNNNN
4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E  NNNNNNNNNNNNNNNN
4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E  NNNNNNNNNNNNNNNN
```

Note: The value of this trace is somewhat limited, other than to determine this type of attack took place, since the snaplength of the capture is set at 150. If we had the entire packet would would have a better indication of what this attack was trying to achieve.

Source of Trace:

My customer's network.

Detect Generated By:

Snort v1.8.3 with the following rule:

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-IIS view source via translate header"; flags: A+; content: "Translate|3a| F"; nocase; reference:arachnids,305; reference:bugtraq,1578; classtype:web-application-activity; sid:1042; rev:3;)

Probability the Source Address was Spoofed:

Medium probability that this packet was crafted. Typical behavior of this exploit establishes a TCP session, making spoofing difficult. In this case we have only 1 packet, and no record of a three way handshake between this host and our host.

Description of the Attack:

This attack is the front door that worms like to use to infect an IIS web server. It is a vulnerability that can be used for a number of purposes, since when it succeeds an attack can run code as the system on the web server.

Attack Mechanism:

The method of attack is a buffer overflow against the Microsoft Indexing Service ISAPI filter. This vulnerability exists on all default Windows NT and Window 2000 machines. When the buffer overflow is successful, the attack can run code of their choosing and perform any system commands on the compromised server.

Correlations:

Dshield.org: http://www.dshield.org/ipinfo.php?ip=24.161.199.95
Shows 1592 records of this IP address against 1152 different addresses.

My NetWatchman:
The correlation from MyNetWatchman.com shows this IP address running various scans across a number of IP address ranges. The full incident log from MyNetWatchman can be accessed via : http://www.mynetwatchman.com/LID.asp?IID=2870631

Incident Id : 2870631	Source Ip : 140.125.31.117

Source Name :	Provider Domain : moe.edu.tw
Net Bios Name :	DNS Name :
Total Event Count : 13	Total Distinct Agent : 4/1400
Response : No Recent Activity	
Status Description : Closed	
Exclusion Reason :	

Network Name/NextNIC	Start IP - End IP
TANET-BNETA/DUMMY	140.117.0.0 - 140.138.255.255

```
NextNIC:99999
Whois provider: moe.edu.tw

Ministry of Education Computer Center (NETBLK-TANET1)
   12th Floor, 106 Hoping East Road
   Section 2
   Taipei, Taiwan, R.O.C
   TW

   Netname: TANET-BNETA
   Netblock: 140.117.0.0 - 140.138.255.255
   Maintainer: MOEC

   Coordinator:
      Chen, Wen-Sung  (WSC1-ARIN)  ZCHEN@TWNMOE10.EDU.TW
      886-2-737-7011

   Record last updated on 30-Sep-1998.
   Database last updated on 16-Aug-2001 23:00:27 EDT.
```

Most Recent Event Date/Time (UTC)	Agent Alias	Agent Type	Log Type	Target Ip	# of IPs Targeted	IP Protocol	Target Port	Port/ Issue Description	Source Port	Explanation	Event Count
18 Feb 2002 20:27:41	jankemi	Perl	Cisco PIX	134.29.x.x	10	6	80	HTTP HTTP Probe	1600	mNW Info	10
18 Feb 2002 05:32:12	foofoo2	win32	NetGear	68.50.x.x	1	6	80	HTTP HTTP Probe	3936	mNW Info	1
17 Feb 2002 19:46:14	esam	Perl	ipchains	129.105.x.x	1	6	80	HTTP HTTP Probe	3287	mNW Info	1

16 Feb 2002 13:23:31	2822	win32	Zone Alarm	208.217.x.x	1	6	80	HTTP HTTP Probe	1204	mNW Info	1

List ALL Incident Activity
Activity Date (UTC): 21 Feb 2002 00:02:02
`Standard escalation email sent to: abuse@moe.edu.tw`

eEye Digital Security:
http://www.eeye.com/html/Research/Advisories/AD20010618.html

Evidence of Active Targeting:

Low, this attacker sent one packet , and didn't do any reconnaissance prior to this attack. In addition the one packet he did send was against a machine that was not vulnerable. Most likely he was scattering his probes across IP ranges to try and be stealthy and not set off port scan alerts.

Severity:

Criticality - The destination machine is my sensor, and temporarily located there for about a week. If they had an IDS system deployed full time and there were attacks against their external sensors, I would rate them higher.
Criticality = 2

Lethality – This machine is a Mandrake Linux box with no web server even installed much less running. This attack does not exploit a vulnerability on this type of OS.
Lethality = 1

System Countermeasures – All non-essential daemons were turned off, tcp wrappers implemented, and ipchains running denying everything.
System Countermeasures = 5

Network Countermeasures – This machine was placed outside the firewall and left to fend for itself.
Network Countermeasures = 0

(Criticality + Lethality) –

(System Countermeasures + Network Countermeasures) = Severity

(1 + 1) – (5+ 0) = -3

Defensive Recommendation:

1. No further defenses needed against this attack for this machine.

Multiple Choice Test Question:

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-IIS ISAPI .ida attempt"; uricontent:".ida?"; no case; dsize:>239; flags:A+; reference:arachnids,552; classtype:web-application-attack; reference:cve,CAN-2000-0071; sid:1243; rev:2;)

The rule above is similar to the one that triggered this alert, what is the difference between the two rules?

A. The rule that triggered the alert came before this rule in the snort.conf file.
B. The rule that triggered the alert came after this rule in the snort.conf file.
C. The rule that triggered the alert fires no matter how large the payload is.
D. This rule will trigger only if the exploit is attempted but not successful.

Answer: C – As explained in my introduction, the snaplength of the capture was set at 150, so it would not be possible for the second rule to fire, as the dsize must be greater than 239 bytes.

Network Detect #5 - Shellcode x86 NOOP

[**] [1:648:4] SHELLCODE x86 NOOP [**]
[Classification: Executable code was detected] [Priority: 1]
02/16-14:41:42.485899 207.77.135.72:3112 -> MY.NET.CLASSC.169:2240
TCP TTL:241 TOS:0x10 ID:45837 IpLen:20 DgmLen:852 DF
AP Seq: 0xE53CEE16 Ack: 0x78560568 Win: 0x2238 TcpLen: 20
[Xref => http://www.whitehats.com/info/IDS181]

[**] [1:648:4] SHELLCODE x86 NOOP [**]
[Classification: Executable code was detected] [Priority: 1]
02/16-14:41:42.617190 207.77.135.72:3112 -> MY.NET.CLASSC.169:2240
TCP TTL:241 TOS:0x10 ID:45847 IpLen:20 DgmLen:852 DF
AP Seq: 0xE53D104E Ack: 0x78560568 Win: 0x2238 TcpLen: 20
[Xref => http://www.whitehats.com/info/IDS181]

Source of Trace:

My customer's network.

Detect Generated By:

Snort v1.8.3 with the following rule:

alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"SHELLCODE x86 NOOP"; content: "|90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90|"; depth: 128; reference:arachnids,181; classtype:shellcode-detect; sid:648; rev:4;)

Probability the Source Address was Spoofed:

Low to nil, the capture shows an actual login to an ftp daemon running on this machine.

Description of the Attack:

Remote buffer overflow attacks frequently use the character code 0x90 as the NOOP character in x86 machine code. This is used as padding for their exploit and increases the chances of success.

Attack Mechanism:

This attack uses the "No Operation" (NOOP) character to provide padding in the payload, and then provide a command and the end of the padding to be executed. A successful attack will overwrite the area in memory allocated to the allowed operation, and the executable code will get run outside of that area and have access to functions it would not have otherwise had access to. This attack signature can be prone to false alert, since a binary file transfer could have 0x90 as part of the payload of a packet legitimately and not be a buffer overflow.

This detect is most likely legitimate, but since I do not have enough information to determine that, I do not want to make any unfounded assumptions. In this detect I am actually more concerned that there is or was an anonymous FTP server running on the company's firewall, than this alert. While that discovery is not part of the Snort alert, it is a by product of fully investigating traffic before coming to any conclusions.

Correlations:

This is not a known bad guy.

Dshield.org: http://www.dshield.org/ipinfo.php?ip=207.77.135.72
Shows 0 records of this IP address against 0 different addresses.

My NetWatchman:
MyNetWatchman.com has no record of this IP address coming up in it's database.

Whois:

UUNET Technologies, Inc. (NETBLK-UUNET1996A)	UUNET1996A	207.76.0.0 - 207.79.255.255
Okidata (NETBLK-OKINET)	OKINET	207.77.134.0 - 207.77.135.255

Further review of all the packets for 207.77.135.72 -> MY.NET.CLASSC.169 shows the following traffic prior to the alert:

```
02/16-14:40:18.092743 MY.NET.CLASSC.169:1453 -> 207.77.135.72:21
TCP TTL:126 TOS:0x0 ID:2057 IpLen:20 DgmLen:48 DF
******S* Seq: 0x784782D3  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

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

02/16-14:40:18.158826 207.77.135.72:21 -> MY.NET.CLASSC.169:1453
TCP TTL:241 TOS:0x0 ID:43565 IpLen:20 DgmLen:44 DF
***A**S* Seq: 0xE5316B77  Ack: 0x784782D4  Win: 0x2238  TcpLen: 24
TCP Options (1) => MSS: 1460

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

02/16-14:40:18.160327 MY.NET.CLASSC.169:1453 -> 207.77.135.72:21
TCP TTL:127 TOS:0x0 ID:2058 IpLen:20 DgmLen:40 DF
***A**** Seq: 0x784782D4  Ack: 0xE5316B78  Win: 0x4470  TcpLen: 20

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

02/16-14:40:21.715951 207.77.135.72:21 -> MY.NET.CLASSC.169:1453
TCP TTL:241 TOS:0x10 ID:43566 IpLen:20 DgmLen:130 DF
***AP*** Seq: 0xE5316B78  Ack: 0x784782D4  Win: 0x2238  TcpLen: 20
32 32 30 20 6E 65 77 73 2E 6F 6B 69 64 61 74 61  220 news.okidata
2E 63 6F 6D 20 46 54 50 20 73 65 72 76 65 72 20  .com FTP server
28 56 65 72 73 69 6F 6E 20 77 75 2D 32 2E 36 2E  (Version wu-2.6.
32 28 31 29 20 46 72 69 20 46 65 62 20 38 20 31  2(1) Fri Feb 8 1
30 3A 34 31 3A 30 37 20 45 53 54 20 32 30 30 32  0:41:07 EST 2002
29 20 72 65 61 64 79 2E 0D 0A                    ) ready...

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

02/16-14:40:21.718826 MY.NET.CLASSC.169:1453 -> 207.77.135.72:21
TCP TTL:127 TOS:0x0 ID:2059 IpLen:20 DgmLen:56 DF
```

```
***AP*** Seq: 0x784782D4  Ack: 0xE5316BD2  Win: 0x4416  TcpLen: 20
55 53 45 52 20 61 6E 6F 6E 79 6D 6F 75 73 0D 0A  USER anonymous..

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

02/16-14:40:21.795684 207.77.135.72:21 -> MY.NET.CLASSC.169:1453
TCP TTL:241 TOS:0x10 ID:43567 IpLen:20 DgmLen:108 DF
***AP*** Seq: 0xE5316BD2  Ack: 0x784782E4  Win: 0x2238  TcpLen: 20
33 33 31 20 47 75 65 73 74 20 6C 6F 67 69 6E 20  331 Guest login
6F 6B 2C 20 73 65 6E 64 20 79 6F 75 72 20 63 6F  ok, send your co
6D 70 6C 65 74 65 20 65 2D 6D 61 69 6C 20 61 64  mplete e-mail ad
64 72 65 73 73 20 61 73 20 70 61 73 73 77 6F 72  dress as passwor
64 2E 0D 0A                                      d...

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

02/16-14:40:21.797131 MY.NET.CLASSC.169:1453 -> 207.77.135.72:21
TCP TTL:127 TOS:0x0 ID:2060 IpLen:20 DgmLen:54 DF
***AP*** Seq: 0x784782E4  Ack: 0xE5316C16  Win: 0x43D2  TcpLen: 20
50 41 53 53 20 49 45 55 73 65 72 40 0D 0A        PASS IEUser@..

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

02/16-14:40:21.911096 207.77.135.72:21 -> MY.NET.CLASSC.169:1453
TCP TTL:241 TOS:0x10 ID:43568 IpLen:20 DgmLen:40 DF
***A**** Seq: 0xE5316C16  Ack: 0x784782F2  Win: 0x2238  TcpLen: 20

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

02/16-14:40:21.913639 207.77.135.72:21 -> MY.NET.CLASSC.169:1453
TCP TTL:241 TOS:0x10 ID:43569 IpLen:20 DgmLen:88 DF
***AP*** Seq: 0xE5316C16  Ack: 0x784782F2  Win: 0x2238  TcpLen: 20
32 33 30 20 47 75 65 73 74 20 6C 6F 67 69 6E 20  230 Guest login
6F 6B 2C 20 61 63 63 65 73 73 20 72 65 73 74 72  ok, access restr
69 63 74 69 6F 6E 73 20 61 70 70 6C 79 2E 0D 0A  ictions apply...
```

=+

There is a CVE under review concerning Anonymous FTP:

CAN-1999-0497 – Anonymous FTP

Evidence of Active Targeting:

There are no prior portscans, and no records of this IP address as a known bad guy. With limited information contained in the trace we need more information to say whether this was an attack or a valid anonymous login. There is a decent probability this was activity completed by the managed service provider, though why they would be using anonymous ftp is a good question. Further research into this will need to take place before being able to determine specifically what happened here.

Severity:

Criticality – This machine is the firewall.
Criticality = 5

Lethality – There are known issues with these types of buffer overflows on Solaris, though I could not find one specifically for the version of ftpd running. At the very minimum it provides reconnaissance information, at worst it allows the firewall to be compromised.
Lethality = 3

System Countermeasures – At this time of this detect there was an an FTP daemon running with anonymous logins enabled.
System Countermeasures = 0

Network Countermeasures – Since this is the firewall, the external interface is completely unprotected.
Network Countermeasures = 0

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

$(5 + 3) – (0 + 0) = 8$

Defensive Recommendation:

1. Turn off the ftp daemon instantly! (Better yet remove it)
2. Implement SSH for any remote management needs, or use the VPN that is in place to manage from internal network interface.
3. Implement tcp wrappers for any services (such as SSH) that must face the external network.
4. Block all inbound FTP if not already blocked.
5. If ftp is going to be enabled and/or installed for any type of access, ensure all available patches are applied.

Multiple Choice Test Question:

If you see a "SHELLCODE x86 NOOP" alert against your firewall, the intrusion analyst should:

A. Start forensics on the target machine.
B. Brief the executive committee of your company.
C. Continue to research and make sure defensive measures are implemented.
D. Nothing.

Answer=C

Executive Summary

This report is an analysis of the provided logs generated by your Snort network intrusion detection system. We will provide as complete an analysis as possible, however since we lack the ability to ask questions of your staff to gain a greater understanding of your computing environment we will have to make some assumptions. Any assumptions we make will be clearly stated as such, so you have the ability to judge the validity of our conclusions. After you review this report, if you wish to have a more complete analysis done, please contact us and we will be happy to provide those services.

This analysis covers a snapshot of the five (5) day time period covered by the logs for February 14, 2002 – February 18,2002. The logs provided were for alerts, scans and out of spec (oos). No binary logs were provided, so this is a limiting factor in the analysis. The ability to go back to a binary file and look at the actual packets that generated the alerts is extremely useful, and can provide information that is not available strictly by viewing the other log files.

The primary tools used for the analysis were SnortSnarf (http://www.silicondefense.com/software/snortsnarf/), perl scripts and shell utilities such as grep, awk, wc,sort and uniq. The majority of the analysis was performed on a linux machine, since that offered the most flexibility in tools to use. For the purposes of making SnortSnarf be able to accurately sort the alerts, all the obfuscated addresses beginning with "MY.NET.xxx.xxx" were changed to "10.10.xxx.xxx" Prior to making this change, there were no instances of IP addresses beginning with "10.10.XXX.XXX", so there was no conflict with any of the other addresses found in the files. In order to avoid confusion on the reader's part, we reversed that substitution when we documented our findings in this report. Since we do not have a copy of the rules at the time these files were generated, we will use the most current Snort rules file, and attempt to extrapolate custom rules based on the outputs found in the log files.

The files used as the input for this analysis were:

alert.020214.gz
alert.020215.gz
alert.020216.gz
alert.020217.gz
alert.020218.gz
oos_Feb.14.2002.gz
oos_Feb.15.2002.gz
oos_Feb.16.2002.gz

oos_Feb.17.2002.gz
oos_Feb.18.2002.gz
scans.020214.gz
scans.020215.gz
scans.020216.gz
scans.020217.gz
scans.020218.gz

Analysis of Alert files:

Here are the detects found in the alert files. All of the alert files were combined into one file, to present the data for the entire period in one summary.

We will be focusing on the most numerous alerts for this analysis. Resolving how to handle the underlying causes of these alerts will yield the greatest return for your security team. We will do the best with the information we have to describe for you what problem the alert is bringing to our attention, as well as some suggested remedies. We will list the remedies with the explanation of the alerts. We strongly recommend a serious evaluation of the remedies suggested, to both mitigate the risk and reduce the current alert frequency. Some very dangerous attacks might be buried in the noise of the other alerts and be missed. If it is decided that some of these alerts are not important to the university, then that signature should be removed from the NIDS so that it will not have the potential to mask other events.

Alert Type	# alerts	# sources	# dests
connect to 515 from inside	137958	143	3
spp_http_decode: IIS Unicode attack detected	68026	142	736
SMB Name Wildcard	60342	264	323
MISC Large UDP Packet [arachNIDS]	33012	28	18
ICMP Echo Request L3retriever Ping	29871	134	14
spp_http_decode: CGI Null Byte attack detected	21972	23	27
SNMP public access	16192	16	146

INFO MSN IM Chat data	11029	106	104
Watchlist 000220 IL-ISDNNET-990517	8710	25	10
High port 65535 udp - possible Red Worm - traffic	7695	158	166

Analysis of the attack alerts:

1. *Connect to 515 from inside:*

We did not locate a rule for this in the standard Snort rule set, but port 515 is usually used by the LPR daemon on multiple flavors of UNIX and Linux. There are known vulnerabilities with many versions of the lpr daemon. Refer to the CERT® Advisory CA-2001-30 (http://www.cert.org/advisories/CA-2001-30.html) for more in depth information on this exploit. Observing the traffic dynamics for this alert we believe that this is most likely legitimate traffic. There were 143 source hosts on the University network that triggered this against 3 destination hosts. The three destination hosts are: MY.NET.150.198, MY.NET.150.205 and MY.NET.1.63. If this traffic was the result of scanning by internal machines looking for port 515 we would expected to see many more destination hosts. Further supporting that theory is the fact that normal lpr traffic has a source port of > 1024, and we did not find any alerts that had a source port of < 1024. We recommend the CCS staff review those machines to validate that lpr traffic is expected. If so, then ensure the patches are up to date for that version as recommended in the CERT® advisory. No other action is required.

The top 5 source hosts:

IP	**# of Alerts**
MY.NET.153.119	10945
MY.NET.88.148	10130
MY.NET.153.114	9834
MY.NET.153.109	6590

The top 3 (only) destination hosts:

IP	**# of Alerts**
MY.NET.150.198	137610
MY.NET.150.205	206
MY.NET.1.63	142

2. *spp_http_decode: IIS Unicode attack detected:*

This alert is generated by Snort's pre-processor http_decode. This is to be able to catch attempts to use Unicode characters in http requests to Microsoft IIS web servers to craft commands that the web server will execute.

The interesting thing about this alert is the majority of the source hosts are inside the university network, and most of the destination hosts are in Korea. We would recommend the CCS security team establish the location of these systems to determine appropriate action. This could be a violation the "Policy for Responsible Computing at UMBC". (http://www.umbc.edu/oit/umbc-aup.html). Defensive measures for any University systems would be to ensure that the latest available patches for IIS have been applied, and watch these alerts to see if new internal hosts appear as targets. If that is the case, it is likely an unauthorized system has been brought online.

The top 5 source hosts:

IP	**# of Alerts**
MY.NET.153.143	4843
MY.NET.153.197	4346
MY.NET.153.106	4214
MY.NET.153.196	2991
MY.NET.153.189	2407

The top 5 destination hosts:

IP	**# of Alerts**
211.115.213.202	5505
211.111.214.125	2226
211.111.220.163	1664
211.233.29.216	1587
211.115.213.207	1488

A search of the KRNIC database at http://whois.krnic.net/ reveals that these IP addresses are registered in Korea:

IP Address : 211.115.213.0-211.115.213.255
Network Name : GNG-IDC-ILOVESCHOOL

Connect ISP Name : GNGIDC
Connect Date : 20001125
Registration Date : 20010621

[Organization Information]
Orgnization ID : ORG215464
Org Name : iloveschool
State : SEOUL
Address : 724 Suseo-Dong Gangnam-Gu
Zip Code : 135-934

[Admin Contact Information]
Name : Yungsuk Cho
Org Name : iloveschool
State : SEOUL
Address : 724 Suseo-Dong Gangnam-Gu
Zip Code : 135-934
Phone : +82-2-538-0629
Fax : +82-2-3420-2301
E-Mail : taiwa@iloveschool.co.kr

[Technical Contact Information]
Name : Yungsuk Cho
Org Name : iloveschool
State : SEOUL
Address : 724 Suseo-Dong Gangnam-Gu
Zip Code : 135-934
Phone : +82-2-538-0629
Fax : +82-2-3420-2301
E-Mail : taiwa@iloveschool.co.kr

3. *SMB Name Wildcard:*

137 is the port used by the NetBIOS SMB Service. This traffic is common among windows (and Linux SAMBA) hosts that are sharing files. It can also be generated by the use of the NBTSTAT command. According to Bryce Alexander in his FAQ from May 10, 2000, this service is used by the 911 bat-chode worm as well. The NBTSTAT command is frequently used for reconnaissance against machines that support NetBIOS. The name of the logged on user, the machine name, the name of the NT domain and the role of the machine in the NT domain can be derived using this command and an IP address. Since the majority of this traffic is going between the same hosts, and all are on the internal network, it is most likely valid NetBIOS traffic. However, there were 2 hosts outside the University that were sources for this alert. Since the only valid use for this service is for LAN file sharing, port 137 should be block into and out of the university's network.

The top 5 source hosts:

IP	# of Alerts
MY.NET.11.6	13473
MY.NET.11.7	12031
MY.NET.11.5	3634
MY.NET.152.163	1170
MY.NET.152.167	666

The top 5 destination hosts:

IP	# of Alerts
MY.NET.11.6	13400
MY.NET.11.7	12003
MY.NET.11.5	3628
MY.NET.152.163	1172
MY.NET.152.167	668

The two external source hosts were 67.32.185.14 and 169.254.22.29. Since these were the only two external sources listed out of a total of 264 sources, we retrieved the address registration information.

67.32.185.14:

Whois from: http://www.geektools.com/cgi-bin/proxy.cgi

Server used for this query: [whois.arin.net]

BellSouth.net Inc. (NETBLK-BELLSNET-BLK12)
301 Perimeter Center North
Atlanta, GA 30346
US

Netname: BELLSNET-BLK12
Netblock: 67.32.0.0 - 67.35.255.255
Maintainer: BELL

Coordinator:
Geurin, Joe (JG726-ARIN) ipadmin@bellsouth.net
678-441-7800 (FAX) 678-441-6968

Domain System inverse mapping provided by:

NS.BELLSOUTH.NET 205.152.0.5
NS.ATL.BELLSOUTH.NET 205.152.0.20

ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE

Record last updated on 28-Feb-2002.
Database last updated on 2-May-2002 19:58:54 EDT.

169.254.22.29:

Whois from: http://www.geektools.com/cgi-bin/proxy.cgi

Server used for this query: [whois.arin.net]

IANA (NETBLK-LINKLOCAL)
Internet Assigned Numbers Authority
4676 Admiralty Way, Suite 330
Marina del Rey, CA 90292-6695
US

Netname: LINKLOCAL
Netblock: 169.254.0.0 - 169.254.255.255

Coordinator:
Internet Corporation for Assigned Names and Numbers (IANA-ARIN) res-ip@iana.org
(310) 823-9358

Domain System inverse mapping provided by:

BLACKHOLE-1.IANA.ORG	192.0.32.18
BLACKHOLE-2.IANA.ORG	192.0.32.19

Record last updated on 12-Oct-2001.
Database last updated on 2-May-2002 19:58:54 EDT.

4. MISC Large UDP Packet:

According to arachNIDS, "This event indicates that an abnormally large UDP packet was sent to your server. This may indicate a denial of service attack or the use of a covert channel." This alert can also be a major source of false positives, as there are several sources of legitimate traffic that use larger UDP packets. The standard Snort rule for this alert sets the size at 4000, which should eliminate most false positives. Some IPSec implementations that use UDP encapsulation, which can be done to allow IPSec to pass through a Network Address translation device, will push the packet size up almost to the MTU size. We recommend reviewing the top destinations, and if it is not expected to have IPSec traffic going to those hosts, then further investigation is warranted. The increase in popularity of VPN's could be a possible explanation for the volume of this traffic observed.

The top 5 source hosts:

IP	**# of Alerts**
216.54.221.197	4759
63.240.15.204	3058
64.152.216.82	2782
63.240.15.199	2716

63.250.205.9	2469

The top 5 destination hosts:

IP	# of Alerts
MY.NET.153.184	10904
MY.NET.153.197	6295
MY.NET.153.171	4759
MY.NET.153.152	2469
MY.NET.152.168	2383

5. *ICMP Echo Request L3retriever Ping:*

The L3 Retriever is a network security assessment tool to discover, map, scan vulnerabilities and audit networks. If this tool is part of the CCS security team's tool kit, then simply checking the source hosts to ensure they are machines used for such official purposes would be sufficient to handle this alert category. We would recommend limiting the source addresses used to perform such scans, so any non-authorized use would stand out. If this tool authorized for use on the network, then further investigation and enforcement of the university's "Policy for Responsible Computing at UMBC." That policy establishes that all computer users shall "Use University computing resources and user accounts only for appropriate University activities." It's interesting to note that the source hosts are well distributed; the destination hosts are targeted at MY.NET.11.6, MY.NET.11.7, and MY.NET.11.5. If these are core servers, then this would be expected traffic, however if the tool is not authorized, then someone has enough information to directly probe your servers for vulnerabilities.

The top 5 source hosts:

IP	# of Alerts
MY.NET.152.163	1157
MY.NET.152.167	676
MY.NET.152.180	624
MY.NET.152.183	616
MY.NET.152.162	614

The top 5 destination hosts:

IP	# of Alerts
MY.NET.11.6	13463
MY.NET.11.7	12012
MY.NET.11.5	3646

MY.NET.5.4	370
MY.NET.150.133	161

6. spp_http_decode: CGI Null Byte attack detected:

This alert is generated by the detection of a NULL byte in the cgi string that is passed to the web server. This is also called the "Poison Null Byte Attack". This vulnerability exploits the differences between the way perl and C interprets that NULL byte. Perl sees the NULL as a valid character, where the system/kernel (written in C) strips it off. This opens up an avenue to bypass erroneous/malicious entry checking in perl and pass an argument to the system to read files or execute programs not intended by the cgi programmer. There is a good explanation of this attack in Phrack Magazine (http://www.phrack.com/show.php?p=55&a=7). This alert can be a false positive if the sites being visited use multi-byte characters such as Simplified Chinese. It is interesting to note that the majority of the alerts come from MY.NET.153.208, going to 209.10.239.135. The whois information for the external IP address is listed below.

The top 5 source hosts:

IP	**# of Alerts**
MY.NET.153.208	14212
MY.NET.153.197	3762
MY.NET.153.184	3641
MY.NET.88.189	98
MY.NET.230.74	85

The top 5 destination hosts:

IP	**# of Alerts**
209.10.239.135	21615
MY.NET.5.96	184
216.33.157.32	39
216.33.88.53	32
64.124.202.25	15

Query: 209.10.239.135
Registry: whois.arin.net
Results:
Globix Corporation (NETBLK-GLOBIXBLK3)

295 Lafayette St- 3rd Fl
NY, NY 10012
US

Netname: GLOBIXBLK3
Netblock: 209.10.0.0 - 209.11.223.255
Maintainer: PFMC

Coordinator:
Hostmaster, Globix Corporation (GCH2-ARIN) arin-admin@GLOBIX.NET
+1-212-334-8500 (FAX) 212.334.8615

Domain System inverse mapping provided by:

Z1.NS.NYC1.GLOBIX.NET 209.10.66.55
Z1.NS.SJC1.GLOBIX.NET 209.10.34.55
Z1.NS.LHR1.GLOBIX.NET 212.111.32.38

ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE

Record last updated on 05-Apr-2001.
Database last updated on 3-May-2002 20:01:14 EDT.

Results brought to you by the GeekTools WHOIS Proxy
Server results may be copyrighted and are used with permission.

7. SNMP public access:

SNMP uses community strings (passwords) to allow access to a network device for configuration changes and polling for reports and statistics. The community string "public" is the default community string for many devices. It is considered a best practice to change that to something unique upon installation of a device to be managed by SNMP. There are many devices that can be enabled for SNMP other than network infrastructure devices, and care should be taken to ensure that SNMP is not enabled where

not intended. In addition, there are several recently disclosed SNMP vulnerabilities (CERT® Advisory CA-2002-03) that call for patching most SNMP agents and taking other defensive measures. Other measures primarily include ingress/egress filtering on your network, and changing the default community strings. More suggestions can be found on the CERT® Advisory page. The Traffic dynamics suggest that ingress/egress filtering is already being leveraged since there are no external source or destination hosts. We strongly recommend investigating these devices and either disable the SNMP agent or change the community string. We applaud the University for being proactive in looking for this even though you have blocked it at the perimeter of the network, and recommend the continuation of this policy.

The top 5 source hosts:

IP	**# of Alerts**
MY.NET.70.177	3826
MY.NET.150.198	2943
MY.NET.150.41	2358
MY.NET.150.245	2244
MY.NET.150.220	2225

The top 5 destination hosts:

IP	**# of Alerts**
MY.NET.152.109	7007
MY.NET.151.114	1336
MY.NET.150.195	1268
MY.NET.150.84	817
MY.NET.5.37	638

8. I*NFO MSN IM Chat data:*

This alert is generated by traffic that appears to be MSN Messenger Instant messenger (IM) traffic. We could find no statement about such traffic in the appropriate usage policies of the university, so we are unsure why this traffic is being monitored. Once possibility is that instant messaging can be a vehicle for the introduction of malicious code into a network, since it bypasses other protection mechanisms and establishes a point to point connection between clients. There was a recent worm named "Cool Worm" that propagated itself through MSN Messenger, and caused malicious JavaScript to be executed. This worm was relatively benign, but the potential is there. We suspect the university is monitoring this as a baseline, to enable a trend to be spotted should the traffic patterns change significantly, or coincide with another incident. To be sure, we checked the whois information for the 64.4.12.0 IP block, and it is registered to MSN. The three observed external addresses are most likely the MSN messenger servers within that block. This would follow the expectations of what this traffic would look like. The best defense against any threats

presented by this are updated anti-virus signatures across the network. We are pleased that the university provides a site license for anti-virus software to all employees, faculty and students, and encourages its use throughout the network.

Whois information for 64.4.12.159 courtesy of http://ws.arin.net/cgi-bin/whois.pl

MS Hotmail (NETBLK-HOTMAIL)
1065 La Avenida
Mountain View, CA 94043
US

Netname: HOTMAIL
Netblock: 64.4.0.0 - 64.4.63.255

Coordinator:
Myers, Michael (MM520-ARIN) icon@HOTMAIL.COM
650-693-7072

Domain System inverse mapping provided by:

NS1.HOTMAIL.COM 216.200.206.140
NS3.HOTMAIL.COM 209.185.130.68

Record last updated on 09-Jan-2001.
Database last updated on 4-May-2002 19:58:14 EDT.

The top 5 source hosts:

IP	**# of Alerts**
64.4.12.159	592
MY.NET.153.45	494
MY.NET.153.108	463
MY.NET.153.109	397
64.4.12.164	360

The top 5 destination hosts:

IP	# of Alerts
MY.NET.153.109	563
MY.NET.153.108	466
64.4.12.159	463
64.4.12.164	406
64.4.12.170	350

9. Watchlist 000220 IL-ISDNNET-990517:

This watchlist appears to be for hosts from the Israeli ISDNNET ISP. Much of this traffic appears to be KaZaA communicating on port 1214. Downloading illegal MP3's and/or setting up an ftp server is a violation of the appropriate usage policies of the university, so blocking on the perimeter of the network could be one way to bring this under control. Beyond being a violation of policy, these peer-peer file sharing programs have been used as a vehicle to introduce Trojans to a machine. In particular, downloading KaZaA has introduced a "client" (their term, many consider it a Trojan) for Brilliant Digital Entertainment's ALTNET. The intent of ALTNET is to use the distributed computing power of all the systems with this code loaded for future unspecified uses. For further information on this issue, reference the paper written by Nicholas Weaver at http://www.cs.berkeley.edu/~nweaver/0wn2.html . We recommend investigating at least the top two destination addresses, since they may be functioning as a "super-node" for KaZaA, which makes them a server for other clients requesting downloads.

The top 5 source hosts:

IP	# of Alerts
212.179.35.118	6130
212.179.66.226	843
212.179.27.176	740
212.179.29.196	417
212.179.35.121	213

The top 5 destination hosts:

IP	# of Alerts
MY.NET.153.150	4746
MY.NET.150.145	1308
MY.NET.153.175	985
MY.NET.152.22	982

MY.NET.153.174 503

10. High port 65535 udp - possible Red Worm – traffic:

There are 7695 alerts of this type in the logs we analyzed. This alert is generated by traffic that is suspected to be the result of the Red Worm (not Code Red), also known as the Adore Worm. This worm affects Linux machines, and is characterized by traffic with a source and destination port of 65535. There is an Adore worm detection and removal tool available via http://www.ists.dartmouth.edu/IRIA/knowledge_base/tools/adorefind.htm. There are a number of machines on the university network that are indicated as compromised by this traffic (104 internal addresses), and we would recommend taking those machines offline immediately and running the afore mentioned tool to investigate further. In addition, since the likelihood of valid traffic having both a source and destination port of 65535, we recommend blocking such traffic at the perimeter of your network. To identify the machines most likely compromised, we isolated all the internal addresses that were sources of the alert, and combined that in a file with all the internal addresses that were destinations of the alert. We then used a combination of the sort, uniq and grep commands to get a list of addresses with 2 instances in the file. (Meaning they were both a source and destination for an alert.)

Possible compromised hosts that should be investigated further are:

MY.NET.149.11,MY.NET.149.23,MY.NET.149.26,MY.NET.149.27,MY.NET.149.28,MY.NET.149.29,MY.NET.149.32,MY.NET.149.43,MY.NET.149.44,MY.NET.149.49,MY.NET.149.50,MY.NET.149.52,MY.NET.149.53,MY.NET.149.57,MY.NET.149.59,MY.NET.149.63,MY.NET.149.64,MY.NET.149.65,MY.NET.149.67,MY.NET.150.83,MY.NET.152.11,MY.NET.152.15,MY.NET.152.157,MY.NET.152.158,MY.NET.152.160,MY.NET.152.161,MY.NET.152.163,MY.NET.152.165,MY.NET.152.166,MY.NET.152.167,MY.NET.152.168,MY.NET.152.171,MY.NET.152.172,MY.NET.152.174,MY.NET.152.175,MY.NET.152.176,MY.NET.152.178,MY.NET.152.179,MY.NET.152.180,MY.NET.152.181,MY.NET.152.182,MY.NET.152.185,MY.NET.152.186,MY.NET.152.20,MY.NET.152.213,MY.NET.152.214,MY.NET.152.215,MY.NET.152.22,MY.NET.152.244,MY.NET.152.245,MY.NET.152.246,MY.NET.152.247,MY.NET.152.248,MY.NET.152.249,MY.NET.152.251,MY.NET.152.44,MY.NET.152.45,MY.NET.152.46,MY.NET.153.140,MY.NET.153.141,MY.NET.153.142,MY.NET.153.145,MY.NET.153.146,MY.NET.153.148,MY.NET.153.157,MY.NET.153.159,MY.NET.153.160,MY.NET.153.161,MY.NET.153.162,MY.NET.153.163,MY.NET.153.164,MY.NET.153.167,MY.NET.153.169,MY.NET.153.172,MY.NET.153.174,MY.NET.153.175,MY.NET.153.176,MY.NET.153.178,MY.NET.153.179,MY.NET.153.181,MY.NET.153.182,MY.NET.153.184,MY.NET.153.186,MY.NET.153.187,MY.NET.153.188,MY.NET.153.190,MY.NET.153.191,MY.NET.153.193,MY.NET.153.197,MY.NET.153.198,MY.NET.153.200,MY.NET.153.202,MY.NET.153.203,MY.NET.153.207,MY.NET.153.209,MY.NET.153.211,MY.NET.253.105,MY.NET.6.48,MY.NET.6.49,MY.NET.6.50,MY.NET.6.52,MY.NET.6.53,MY.NET.6.60,MY.NET.88.148

The top 5 source hosts:

IP	**# of Alerts**

MY.NET.6.49	2168
MY.NET.6.52	1653
MY.NET.6.48	1387
MY.NET.6.50	1322
MY.NET.6.60	134

The top 5 destination hosts:

IP	**# of Alerts**
MY.NET.152.47	241
MY.NET.153.146	212
MY.NET.153.150	189
MY.NET.152.157	170
MY.NET.153.207	147

We also compiled some statistical analysis of the scans that were documented in the log files. Scans are usually an attempt at reconnaissance, and a pre-cursor to attack. While getting scanned is a normal part of being connected to the internet, it is wise to baseline the "expected" scanning. This allows analysts to spot an increase in certain types of scans, and possibly head off an attack, compromise or information leakage before it occurs.

NOTE: We would like to thank Christof Voemel for his outstanding scripts that allowed us to draw some statistics most efficiently from the data.

The top 5 scan types are:

# of scans	Type
2073421	UDP
393819	SYN ******S*
19095	VECNA ****P***
255	NULL ********
15	UNKNOWN *2*A**S* RESERVEDBITS

This only gives one side of the picture however, because it does not indicate what was being scanned for, only the method of scanning. Some scans are attempts at fingerprinting; others target specific ports to see if they are open and potentially vulnerable to attack.

The top 10 scanned ports:

# of scans	port# and probable application*
307152	80 - http
28329	1214 - KaZaA
24156	139 – Netbios Session Service
5633	135 – RPC Location Service (used with Windows)
5350	1026 – remote_login – network terminal (nterm)
5040	389 - LDAP
4915	443 - HTTP protocol over TLS SSL
4033	21 – FTP Control
1842	5190 – AOL Instant Messenger
1454	28204 - Unknown

- Explanation of what the port is used for is based on the most probable application. Many Trojan application use ports registered to other applications to disguise themselves as valid traffic on the network.

The top 10 sources for scans are:

Count	**Source Address:**
460502	MY.NET.60.43
151242	MY.NET.6.49
109494	MY.NET.6.52
108583	MY.NET.6.45
99473	MY.NET.6.48
94580	MY.NET.6.50
61752	MY.NET.60.11
45919	MY.NET.6.60
41015	MY.NET.6.53
20039	205.188.228.17

The top 10 targets for scans are:

Count	**Destination**
111019	MY.NET.1.7
89166	MY.NET.1.3
61397	MY.NET.1.4
50429	MY.NET.6.45
46141	MY.NET.11.6
42714	MY.NET.11.7

41754	MY.NET.60.43
29228	MY.NET.6.60
26726	MY.NET.60.11
25469	MY.NET.153.207

All of the OOS packets originated outside the university network. Most of the traffic had a destination port of 1214, which as discussed previously is associated with KaZaA. Port 113 was also a popular port, that is used by identd, so if most probably an attempt at reconnaissance. For defensive measures we recommend the university block packets that are out of spec at the perimeter, and do not allow them into or out of the network.

Here is a summary of the OOS data:

02/14-01:36:19.064086 217.85.238.101:1063 -> MY.NET.150.145:1214
02/14-13:05:51.965357 134.176.202.100:1499 -> MY.NET.150.133:1214
02/14-19:24:17.351038 62.103.232.167:2810 -> MY.NET.150.133:1214
02/14-19:45:43.336808 62.103.232.167:1706 -> MY.NET.150.220:1214
02/14-19:56:44.002379 62.103.232.167:3015 -> MY.NET.150.133:1214
02/14-20:23:29.497373 62.103.232.167:3215 -> MY.NET.150.133:1214
02/15-12:44:33.691117 216.218.255.227:45248 -> MY.NET.152.183:113
02/15-14:38:36.168403 216.218.255.227:47231 -> MY.NET.152.183:113
02/15-14:46:21.678074 216.218.255.227:47373 -> MY.NET.152.183:113
02/17-09:57:38.319466 142.177.114.186:1074 -> MY.NET.150.133:1214
02/17-10:52:10.152235 217.208.152.26:64993 -> MY.NET.150.133:1214
02/17-23:45:11.856609 68.55.20.174:1230 -> MY.NET.5.96:80
02/18-07:47:26.210924 95.0.100.57:3282 -> MY.NET.150.133:1214
02/18-10:51:26.448356 209.86.166.2:32945 -> MY.NET.150.133:1214
02/18-10:51:29.414118 209.86.166.2:32945 -> MY.NET.150.133:1214
02/18-10:51:35.420940 209.86.166.2:32945 -> MY.NET.150.133:1214
02/18-10:52:19.636743 209.86.166.2:32962 -> MY.NET.150.220:1214
02/18-13:24:15.049695 216.53.71.65:44765 -> MY.NET.5.243:113
02/18-13:34:07.179877 216.53.71.65:47771 -> MY.NET.5.243:113
02/18-16:50:43.229921 24.226.53.14:2601 -> MY.NET.150.133:1214

The top 10 talkers for all logs are:

Count	Address
460502	MY.NET.60.43
151242	MY.NET.6.49
137622	MY.NET.150.198
111019	MY.NET.1.7
109494	MY.NET.6.52
108583	MY.NET.6.45
99473	MY.NET.6.48
94580	MY.NET.6.50
89166	MY.NET.1.3
61752	MY.NET.60.11

Since all of the top talkers were internal hosts, we also thought it useful to look at the top external talkers.

The top 10 external talkers for all logs:

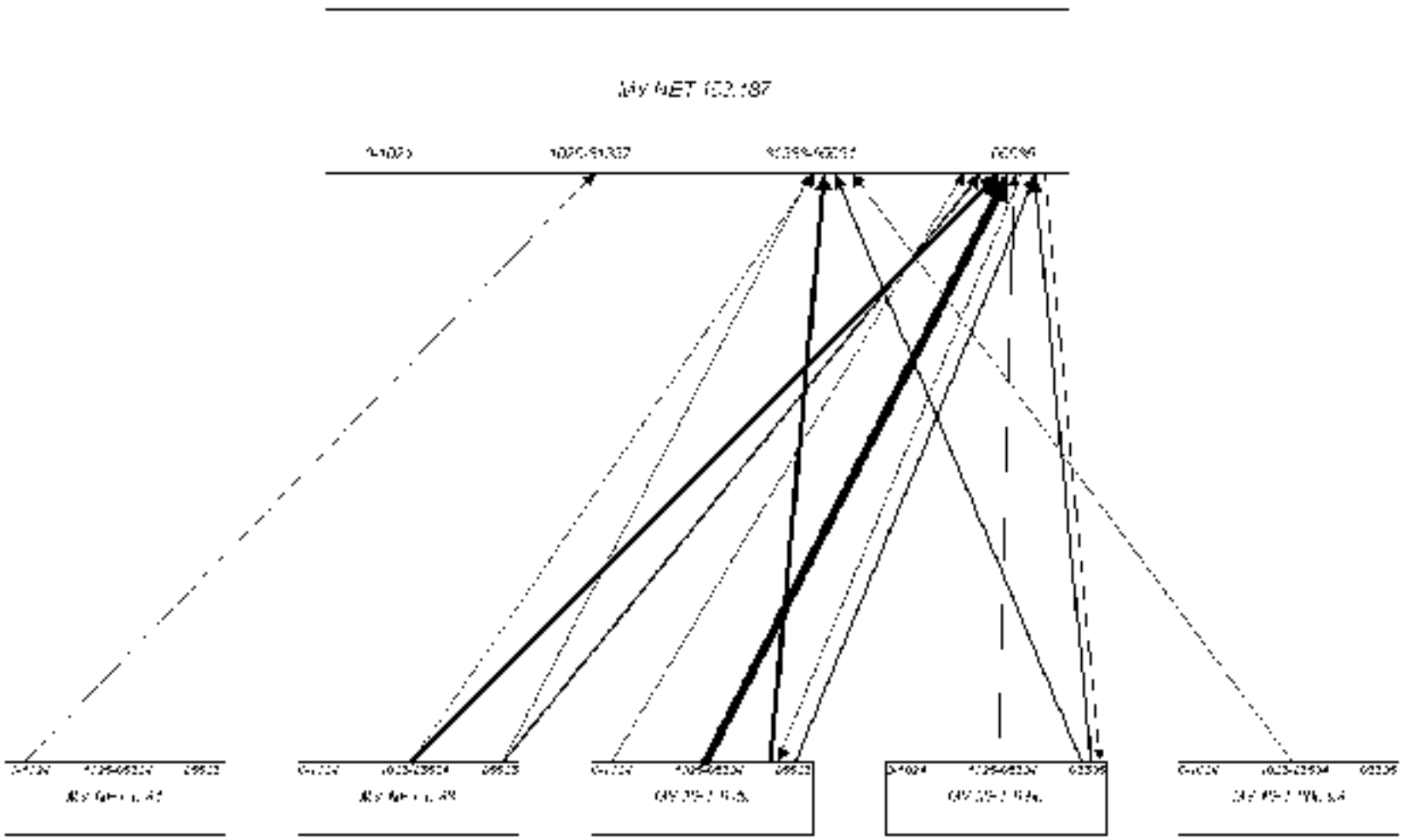
Count	Address
21615	209.10.239.135
20039	205.188.228.17
13948	205.188.228.33
13106	63.215.70.141
11029	205.188.228.1
10982	205.188.228.65
9126	205.188.228.17
8525	205.188.228.65
6130	212.179.35.118
6041	131.118.254.38

Port Link Diagram:

The host MY.NET.153.187 seemed very popular in the alerts concerning trojan/worm activity. Since trojans and worms represent significant risk and can spread rapidly, we thought it would be useful to diagram some of this traffic with the other hosts that appeared in those alerts to see what we could determine. We can see that there is quite a lot of traffic using port 65535 as a destination on MY.NET.153.187. This suggests that this host is most likely compromised with the Adore Worm. Once the worm infects a host, it establishes a back door on port 65535. After the back door is established, attackers can telnet to that port and gain root access. Unless

you have a custom, networked application that uses port 65535 for the server component, this diagram shows the popularity of that port for traffic. Potential Back Orifice traffic, as well as WinVNC traffic was also noted for this host, so a complete analysis of this machine should be completed prior to allowing it back on the network, to ensure that no other compromises or inappropriate applications exists.

Port Link Diagram for MY.NET.153.187



For additional confirmation of that analysis, we looked at all alerts with a destination host of MY.NET.153.187 and counted the destination ports. The pattern reflected in the diagram above is consistent with what we found.

Summary of Destination ports from Alerts
where MY.NET.153.187 is the destination host.

Count	Port #
51	65535
4	20712
3	65280
3	1028
2	8080
2	65532
2	3128
2	1528
2	137
1	65529
1	65408
1	61439
1	35717
1	31337
1	1158
1	1099

Summary

In conclusion, we have used the information provided by your security team to provide some insight into some of the types of events you are experiencing on your network. While there are some limitations to our analysis because of our lack of knowledge of your environment, we believe you will find this represents a good start in getting a handle on your network security posture. We reviewed the resources available on the university's internal web sites concerning security education, current threats and the policies of the

university. While not restricted to specifically those threats that we addressed as part of the analysis we were pleased with the comprehensiveness of the resource. The university appears to be taking a holistic approach to information security and addressing the need to both establish policies and provide educational resources for its users. We also reviewed past analyses of your environment, and the types and number of alerts suggests that the university security team is improving the signature set, making configuration improvements and actively trying to take compromised hosts off the network. We believe this to be one of the most important factors in effective intrusion detection, and applaud the university's security team for their efforts.

References

1. Baldwin, Lawrence, "How myNetWatchman works.", URL: http://www.myNetWatchman.com (5 May 2002).

2. Cherbaka, Phillip,"GCIH Practical Assignment V2.0",(2 December 2001), URL: http://www.giac.org/practical/Phillip_Cherbaka_GCIH.doc , (10 May 2002).

3. URL: http://packetstorm.decepticons.org/0010-exploits/iis-unicode.txt, (10 May 2002).

4. "Internet Security Systems Security Alert #30", 6 July 1999, URL: http://xforce.iss.net/alerts/advise30.php , (10 May 2002).

5. "CAIDA Analysis of Code-Red", http://www.caida.org/analysis/security/code-red/ , (10 May 2002).

6. Maiffre, Marc, & Permeh, Ryan, ".ida "Code Red" Worm", 16 July 2001, URL: http://www.eeye.com/html/Research/Advisories/AL20010717.html , (10 May 2002).

7. "arachNIDS Archive",URL: http://activeworx.com/info/ (10 May 2002).

8. Voemel, Christof, "GCIA Practical Assignment Version 3.0", 7 September 2001, http://www.giac.org/practical/Christof_Voemel_GCIA.txt , (10 May 2002).

9. Hrabowski, Freeman A. , "POLICY FOR RESPONSIBLE COMPUTING AT UMBC", 26 September 1996, http://www.umbc.edu/oit/umbc-aup.html , (10 May 2002).

10. "ResNet Guidelines for Acceptable Use",URL: http://resnet.umbc.edu/resnet_guidelines.html , (10 May 2002).

11. "CERT® Advisory CA-2001-30 Multiple Vulnerabilities in lpd", 15 Nov. 2001, URL: http://www.cert.org/advisories/CA-2001-30.html ,(10 May 2002).

12. Alexander, Bryce, "Port 137 Scan", 10 May 2000, URL: http://www.sans.org/newlook/resources/IDFAQ/port_137.htm , (15 April 2002).

13. Lisa Phifer, "Slipping IPSec past NAT", 19 April 2001, URL: http://www.isp-planet.com/technology/2001/ipsec_nat.html ,(1 May 2002).

14. "Retriever Datasheet",http://www.safecomms.com/products/symantec/retriever_ds.html ,(10 May 2002).

15. Phrack Magazine, "Perl CGI problems", Vol. 9 , Issue 55, 07 of 19, 09 Sep. 1999, URL: http://www.phrack.com/show.php?p=55&a=7 ,(8 May 2002).

16. "Multiple SNMP Vulnerabilities", 12 February 2002,URL: http://www.counterpane.com/alert-snmp.html ,(10 May 2002).

17. Poulsen, Kevin, "MSN Messenger Worm Entices the Unwary",13 Feb 2002, URL: http://online.securityfocus.com/news/331 ,(05 May 2002)

18. Leyen, John, "Not Brilliant, KaZaA's crackers", 8 April 2002,URL: http://www.theregister.co.uk/content/6/24761.html ,(7 May 2002).

19. Fearnow, Matt, & Stearns, William, "Adore Worm Detection and Removal Tool", URL: http://www.ists.dartmouth.edu/IRIA/knowledge_base/tools/adorefind.htm , (10 May 2002).

20. Poor, Mike, "GCIA Practical Assignment, v3.0", URL: http://www.giac.org/practical/Mike_Poor_GCIA.doc ,(2 May 2002).

21. Farshchi, Jamil, "GCIA Practical Assignment v3.0", URL: http://www.giac.org/practical/Jamil_Farshchi_GCIA.doc ,(8 May 2002).

22. Kuethe, Chris, "GCIA Practical Assignment", URL: http://www.giac.org/practical/chris_kuethe_gcia.html ,(10 May 2002).

23. Goodwin, PJ, "GIAC Level Two Intrusion Detection in Depth", URL: http://www.giac.org/practical/PJ_Goodwin_GCIA.doc , (7 May 2002).

24. Fielding, Et al, "RFC2616", Hypertext Transfer Protocol -- HTTP/1.1, (1999), URL: http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html ,(10 May 2002).

25. "Patch Available for 'Web Server Folder Traversal Vulnerability", Microsoft Security Bulletin MS00-78, (17 Oct. 2000), URL: http://www.microsoft.com/technet/security/bulletin/MS00-078.asp , (10 May 2002).

26. "Patch Available for 'File Permission Canonicalization' Vulnerability", Microsoft Security Bulletin MS00-57, (10 Aug. 2000), URL: http://www.microsoft.com/technet/security/bulletin/ms00-057.asp, (10 May 2002).

27. “Snort Signatures Database – Hack a Tack Trojan”, (13 March 2002), URL: http://www.snort.org/snort-db/sid.html?id=614 , (10 May 2002).

28. “CAN-1999-0660”, URL: http://cve.mitre.org/cgi-bin/cvename.cgi?name=1999-0660 ,(10 May 2002).

29. Hypponen, M.,Erdelyi, G., & Rautianinen, S.,“F-Secure Virus Descriptions-Code Red”,(August 2001), URL: http://www.europe.f-secure.com/v-descs/bady.shtml , (10 May 2002).

30. “Unchecked Buffer in Index Server ISAPI Extension Could Enable Web Server Compromise”, Microsoft Security Bulletin MS01-033, (18 Jun. 2001), URL: http://www.microsoft.com/technet/security/bulletin/ms01-033.asp ,(10 May 2002).

31. “'Relative Shell Path Vulnerability”, Microsoft Security Bulletin MS00-052, (28 July 2000), URL: http://www.microsoft.com/technet/security/bulletin/MS00-052.asp ,(10 May 2002).

32. “All versions of Microsoft Internet Information Services Remote buffer overflow (SYSTEM Level Access)”, (18 Jun. 2001), URL: http://www.eeye.com/html/Research/Advisories/AD20010618.html ,(10 May 2002).

33. “CAN-1999-0497”, URL: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0497 ,(10 May 2002).

34. “CERT® Advisory CA-2002-03 Multiple Vulnerabilities in Many Implementations of the Simple Network Management Protocol (SNMP)”, (28 Apr. 2002), URL: http://www.cert.org/advisories/CA-2002-03.html ,(10 May 2002).

37. Weaver, Nicholas, “Reflections on Brilliant Digital: Single Points of Internet 0wnership”, URL: http://www.cs.berkeley.edu/~nweaver/0wn2.html ,(10 May 2002).