Global Information Assurance Certification Paper

GCIA Practical Assignment, v3.3

Intrusion Detection In Depth

Richard A. Baker

November 4, 2002

PART 1 – DESCRIBE THE STATE OF DETECTION (30 points)
Choose an attack, reconnaissance technique, denial of service, or exploit

LET'S TAKE A LOOK AT "NIMDA"

Background

All sniffer information presented for this analysis was captured using a government network intrusion detection system (IDS) referred to as Joint Intrusion Detection (JID), software version 2.3.1, and displayed using a front-end browser developed by the Department of Defense called a Joint Analysis Browser (JAB). This method of data collection is one of several deployed by the Department of Defense. The intrusion detection system data presented here has been sanitized to include Internet protocol addresses, user accounts, and date/time stamps. This attack was acquired from a United States Department of Defense site located at Scott Air Force Base, Illinois.

Nimda was selected to fill the requirements for Part 1 of this practical because of its uniqueness. Nimda is unique in that it is not only the first worm to modify existing web sites so as to start offering infected files for download, but it is also the first worm to use normal end user machines to scan for vulnerable web sites. The use of these machines allows Nimda to connect with intranet web sites behind their firewalls - Code Red couldn't do this (http://www.f-secure.com/v-descs/nimda.shtml).

Attack

Known as both a complex computer virus and worm, Nimda (admin spelled backwards) arrived in the summer of 2001 and swept like a tornado across the Internet, attacking desktops and servers running Microsoft's Windows 95, 98, Me, NT 4, and 2000 operating systems. No one is sure exactly where it originated (China is heavily favored), but we can assume the initial (first) attack was targeted. How or why its target(s) were selected we may never know. One thing we do know however, an initial site had to be selected and attacked, and once that first site was infected Nimda began working its magic, spreading itself across the Internet. This fast-spreading hybrid (virus/worm), "a combination of Code Red and Apost" (http://www.winplanet.com/winplanet/tips/3787/1), left many a system administrator pulling his/her hair out over the seemingly never-ending chore of patching and re-patching their Web-based software.

Nimda exploits the dozens of known vulnerabilities in the Microsoft Web server, Outlook mail client, and Internet Explorer browser. Symantec Corp. refers to Nimda as "a denial-of-service attack tool like Code Red II, scanning for corporate subnets" (http://www.f-secure.com/v-descs/nimda.shtml). It does its damage by worming its way into Internet Information Server (IIS) Web pages, infecting PCs through vulnerable Internet Explorer browsers, corrupting files, and scanning IP addresses for more victims to infect. In short, Nimda is a very

aggressive virus/worm capable of generating large volumes of network and Internet traffic which in turn, chokes and slows both down.

Gaining Access – "The Exploit"

Nimda arrives on scene in various fashions. Email is one common way for your system to become infected. In this fashion, Nimda arrives as an email attachment named readme.exe. Once the attachment is opened the virus/worm uses Mailing API (MAPI) to access the PC's email address book and propagates itself by sending emails to all the addresses. Once you have an infected PC on your network, it can in turn infect your web site. This is done through the user's infected local html files by copying or uploading infected pages to your Web server. Conversely, infected web servers can infect local users if file sharing is allowed.

Another way to become infected by Nimda is via the Internet. In this method, not only does Nimda exploit the buffer-overflow vulnerability, as did Code Red and its variants, but it also exploits a flaw in IIS called the Unicode Directory Traversal Vulnerability. Using an infected IIS WEB server, Nimda searches for other vulnerable IIS servers. In many cases an infected Web server may display a Web page prompting unsuspecting visitors to download an infected file. Once downloaded, Nimda can spread via the user's PC through email propagation.

The actual lifecycle of Nimda can be split in to four parts: infecting files, mass mailing, Web worm, and LAN propagation. The four-part lifecycle and explanations listed below were taken from F-Secure Virus Descriptions at http://www.f-secure.com/v-descs/nimda.shtml.

1) File infection: Nimda locates EXE files from the local machine and infects them by putting the file inside its body as a resource, thus 'assimilating' that file. These files then spread the infection when people exchange programs, such as games.

2) Mass mailer: Nimda selects and sends itself to addresses in the HTML files. It also uses MAPI to locate email addresses in the users inbox, sending itself to each address. Each mailing contains the README.EXE attachment which once opened activates the worm/virus.

3) Web worm: Nimda scans the Internet looking for vulnerable www servers. Once such a web server is found, the worm tries to infect it by using several known security holes. If successful, the worm may modify random web pages on the site by adding a tiny javascript. When the infected page is accessed the script opens a browser window referencing a file called README.EML, which may automatically download and execute the virus on the users machine. It is possible that Web surfers visiting the site may automatically become infected without their knowledge.

4) LAN propagation: Nimda spreads itself across the local network through

shared drives and opens new shares on infected machines. Once a machine is infected, it writes a hidden file named RICHED20.DLL to directories containing extension .DOC, .NWS, and .EML files. Using these extensions Nimda than copies itself to the directory. Anyone opening these files will execute RICHED20.DLL and in so doing become infected.

Analysis

On reviewing the 18 September 2002, 09:00 zulu (Z) Joint Intrusion Detection / Joint Analysis Browser Log I came across a Nimda alert captured at 0932Z by Joint Intrusion Detection. The below data is a classic example of a Nimda exploit for the Directory Traversal vulnerability.

All Nimda attacks begin by completing the basic TCP 3-way handshake thus clearing the way for the exchange of data via destination port 80. After completion of the 3-way handshake the attacker sends a script of constructed URL commands using the "../" designed to move within the destination server. You can see from the below streamed data script that among the commands sent is a "/cmd.exe?", accessing the command prompt and "/c+dir" designed to allow the attacker to view the contents of the hard drive. Had these commands been successful, the way would be clear for Nimda to complete the data transfer.

Source Data:
=== Intruder Script from Stream File "NIDabc1-020918.10.11.stream.init" ===

IP Header from first packet:
Ethernet source : 0:60:47:12:e9:20
Ethernet destination : 0:3:47:70:dd:6d
Ethernet bytes : 82
Ethernet time : Wed Sep 18 09:32:19 2002
Network protocol : IP
Network source : Black.hat (Unknown)
Network destination : DoD.site (Unknown)
Network bytes : 68
Transport protocol : tcp
Transport bytes : 28
Application source : 1505
Application destination : 80
NIT total length : 102
NIT message length : 94

--- The stream script ----------

GET /_vti_bin/..%255c../..%255c../..%255c../winnt/system32/cmd.exe?/c+dir
HTTP/1
.0
Host: www

Connnection: close

=== End of Intruder Script from Stream File "NIDabc1-020911.10.11.stream.init" ===
Destination Data:
=== Intruder Script from Stream File "NIDabc1-020918.10.11.stream.dest" ===

IP Header from first packet:
Ethernet source : 0:3:47:70:dd:6d
Ethernet destination : 0:60:47:12:e9:20
Ethernet bytes : 80
Ethernet time : Wed Sep 18 09:32:19 2002
Network protocol : IP
Network source : DoD.site (Unknown)
Network destination : Black.hat (Unknown)
Network bytes : 66
Transport protocol : tcp
Transport bytes : 24
Application source : 80
Application destination : 1505
NIT total length : 100
NIT message length : 92

--- The stream script ----------

HTTP/1.1 403 Forbidden
MIME-Version: 1.0
Server: Simple, Secure Web Server 1.1
Date: Wed, 18 Sep 2002 09:27:52 GMT
Connection: close
Content-Type: text/html
….
….
….

=== End of Intruder Script from Stream File "NIDabc1-020918.10.11.stream.dest" ===

The following are some examples of scripts (FootPrint) Nimda produces for any web server listing on tcp/port 80:

GET /scripts/root.exe?/c+dir
GET /MSADC/root.exe?/c+dir
GET /c/winnt/system32/cmd.exe?/c+dir
GET /d/winnt/system32/cmd.exe?/c+dir

GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir
GET /_mem_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir

GET /_vti_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir
GET /msadc/..%5c../..%5c../..%5c/..\xc1\x1c../..\xc1\x1c../..\xc1\x1c../
winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc0/../winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc0\xaf../winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc1\x1c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc1\x9c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%2f../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%35c../winnt/system32/cmd.exe?/c+dir

Nimda can be scripted to exploit IIS systems previously compromised by Code Red or Code Red II worms. The first four above entries are scripted to allow the attacker access to the back door left open by Code Red II, the remaining log entries are examples of Directory Traversal vulnerability exploits (see CERT® Advisory CA-2001-26 Nimda Worm at http://www.cert.org/advisories/CA-2001-26.html).

An exploit of Nimda that has not been overly publicized or talked about is that Nimda, in some instances, will execute the addition of a guest account to the administrator's group. What this means is that a user "guest" has access to this machine with no password required. Although not all machines with "Open Guest Share" are infected, this is a possible symptom/infection, and a grave security violation to boot. GFI, a leading worldwide developer of messaging, content security and network security software, in its description of Nimda at http://www.gfi.com/news/press.asp?release=nimdaworm&lcode=en briefly addresses this issue under the heading "If you have been infected." stating "The guest account is added to the administrator's group."

Prevention

Below are five basic (common sense type) tips taken from http://www.winplanet.com/winplanet/tips/3787/3/ for protecting your system(s) against the Nimda Worm and eliminating it in the event you do get the infection.

Tip 1. Become an Educated User

Detailed information regarding the Nimda worm can be found in CERT Advisory CA-2001-26, from the Computer Emergency Response Team Coordination Center at Carnegie Mellon University.

Tip 2. Get and Use Anti-virus Software

Anti-virus software is a must-have. Get it, install it, and use it regularly to help keep your system clean of viruses and worms.

Tip 3. Update Anti-virus Software

Regularly updating anti-virus software will protect your system from new viruses and worms.

Tip 4. Delete Suspicious Messages

Always scan your email and attachments for viruses before opening, and fix or delete them if they're infected. If you suspect an email is infected, play it safe and delete it.

Tip 5. Download Microsoft Patches

Free patches are available, just go to http://www.microsoft.com/downloads/search.asp and search using key word "Nimda" and select your operation system.

Summary

We have just taken a brief, but not overly comprehensive look at the hybrid worm/virus called Nimda. We've learned that while the initial attack may well have been targeted, subsequent infections are the result of exposure to infected machines/systems through direct or indirect contact. Exploiting vulnerabilities in Microsoft operating systems and programs, Nimda spreads itself by way of file infection, Lan propagation, mass email, and Website infection. While Nimda is an extremely aggressive worm capable of causing bandwidth denial-of-service conditions on networks with infected machines, the threat it poses can be negated through user education, Microsoft patches, and the use anti-virus software. To learn more about Nimda, check out the references listed below or just type, "Nimda, How it works" into your favorite search engine and see what can be found.

References

F-Secure Virus Descriptions, "Nimda"
URL: http://www.f-secure.com/v-descs/nimda.shtml

Ellen Messmer and John Fontana – Network World Fusion, Update: "Nimda worm spreads three ways; seen as major threat"
URL: http://www.nwfusion.com/news/2001/0918admindll.html

Ellen Messmer and John Fontana - Network World Fusion, "Nimda virus riles up Microsoft users"
URL: http://www.nwfusion.com/news/2001/0924nimdams.html

D. E. Levine - WinPlanet, “Handling Nimda”
URL: http://www.winplanet.com/winplanet/tips/3787/1
URL: http://www.winplanet.com/winplanet/tips/3787/2

Dean Chetkovich - California State Polytechnic University, “NIMDA Virus Warning”
URL: http://www.csupomona.edu/~housing/update102.htm

Robert Vamosi - California State Polytechnic University, “How it works (Nimda)”
URL: http://www.csupomona.edu/~housing/update102.htm

Robert Lemos - California State Polytechnic University, “Lethal worm spells double trouble”
URL: http://www.csupomona.edu/~housing/update102.htm

Carnegie Mellon CERT Coordination Center, “CERT® Advisory CA-2001-26 Nimda Worm”
URL: http://www.cert.org/advisories/CA-2001-26.html

Norman - The Antivirus Company, “W32/Nimda.A@mm”
URL: http://www.norman.no/virus_info/w32_nimda_a_mm.shtml

GFI Software Ltd, “NEWS/Nimda worm: Description”
URL: http://www.gfi.com/news/press.asp?release=nimdaworm&lcode=en

PART 2 – NETWORK DETECTS (30 points)
Submit three network detects, with detailed analysis

Detect 1: ACK Flood Attack

The following extracts show the beginning and ending of what I presume to be a weak attempt at an ACK flood. I say weak because even though the attack runs for 21 hours, the sheer number of attempts falls far short of what would be required to achieve and maintain a denial of service. Perhaps this is a beginner just testing the waters. To save space ACK flood data between 00:55 and 21:36

is not posted.

00:00:55.134488 46.5.180.250.61013 > 64.154.80.51.80: P
555149661:555151121(1460) ack 1599014187 win 33580 [tos 0x10]
00:00:55.244488 46.5.180.250.61013 > 64.154.80.51.80: P
043873286:1043874203(917) ack 3251102771 win 17520 (DF)
00:00:55.344488 46.5.180.250.61013 > 64.154.80.51.80: P 0:4380(4380) ack
7301 win 33580 [tos 0x10]
00:00:55.354488 46.5.180.250.61013 > 64.154.80.51.80: P 0:2377(2377) ack
9678 win 33580 [tos 0x10]
- - -
- - -
- - -
21:36:33.184488 46.5.180.250.62991 > 64.154.80.51.80: P
2303435203:2303436341 (1138) ack 3185377023 win 17520 (DF)
21:36:33.564488 46.5.180.250.62991 > 64.154.80.51.80: P
881943846:881944983(1137) ack 3413026616 win 33580 [tos 0x10]
21:36:47.684488 46.5.180.250.63005 > 64.154.80.51.80: P
2307731340:2307732540 (1200) ack 3719953110 win 17520 (DF)
21:36:48.294488 46.5.180.250.63005 > 64.154.80.51.80: P
1412224102:1412225301 (1199) ack 2882746728 win 33580 [tos 0x10]

Source of Trace:

Posted on http://www.incidents.org/logs/Raw, file 2002-5-17.17.

Detect was generated by:

The raw data taken from incidents.org was run through tcpdump version 3.6 for analysis.

Probability the source address was spoofed:

High. Evidence suggests the source address is spoofed. A scan for open ports identified IP 46.5.180.250 as “not an active host.” A resolution of the source address revealed the following:

Search results for: 46.5.180.250
OrgName: Internet Assigned Numbers Authority
OrgID: IANA

NetRange: 46.0.0.0 - 46.255.255.255
CIDR: 46.0.0.0/8
NetName: RESERVED-46
NetHandle: NET-46-0-0-0-0
Parent:
NetType: IANA Reserved
Comment:

RegDate:
Updated: 2002-08-23

Running both RIPE Whois (Europe) and APNIC Whois (Asia) yielded the same IANA Reserved resolution. (Note: Most DoS attacks are spoofed and IANA reserved IP ranges are often used.)

Description of attack:

The above attack shows the source IP attacking the destination IP 64.154.80.51 over and extended period of time (21 hours). This IP would most likely be selected because the attacker believes this remote system does not exist and that the firewall will allow the ACK to be added to the connections table. If the attacker has done his home work correctly, there will be no remote system to send a RST or FIN to break this connection. This results in each ACK packet receiving a 1-hour time out on the host's system, thus filling the connections table causing a denial of service.

Attack mechanism:

The ACK flood starts a TCP connection with the ACK packet. If the destination firewall rule set allows other than TCP SYN initial connection attempts, the source connection is added to the connections table. Once added to the connections table it is given a 3600 second (1 hour) time out. If the source continues with this attack, the connections table soon fills up resulting in a destination denial of service.

Correlations:

A search of DShield.Org for IP 46.5.180.250 yielded no results. However, the ACK flood is a well documented denial of service attack and is referenced in "Distributed Denial of Service Attack, Is There really a Threat" by David Dittrich of the University of Washington (http://staff.washington.edu/dittrich/talks/sec2000.ppt), CERT® Incident Note IN-2000-05, dated May 2, 2000 (<http://www.cert.org/incident notes.IN-2000-05.html>) and "Understanding the FW-1 State Table" by Lance Spitzner (http://www.collusion.org/Article.cfm?ID=307)

Evidence of activity targeting:

High. Evidence suggests active targeting. This attack is from a single source IP 46.5.180.250 against IP 64.154.80.51/port 80 and lasts for 21 hours.

Severity:

Criticality = 5 (A properly executed Ack flood is a targeted denial of service (DoS) attack. The target of a DoS attack should immediately seek answers to these three questions as a minimum: 1. Where my defenses up to the challenge? 2. Why am I being attacked? 3. Who is attacking me?)

Lethality = 4 (attack has the potential to cause a denial of service and could even be an initial probe for a heaver attack to follow)

System Countermeasures = 4 (it is assumed this is a modern operating system, all patches, added security such as tcp wrappers and secure shell, etc)

Network Countermeasures = 5 (a validated restrictive firewall is assumed, only one way in or out)

Severity = [(5+4) – (4+5)] = 0

Defensive recommendation:

Blocking this IP address is a reasonable course of action. Better yet, blocking all IANA reserved IP ranges would be more appropriate. It would also be appropriate to deny any TCP packets that are not initiated by the proper TCP three-way handshake and/or that contain illegal TCP flag combination

Multiple choice test question:

An ACK attack, like a SYN flood is intended to:

A. Create a denial of service
B. Gather information
C. Gain root directory access
D. Spread malicious logic

Correct response: A

Posting of Analysis:

After posting my analysis on Oct 10, 2002 I received the following email. I will attempt to answer Gary Morris' questions below.

From: Gary Morris [mailto:gmorris@govolution.com]
Sent: Thursday, October 10, 2002 10:05 AM
To: intrusions@incidents.org

Subject: RE: LOGS: GIAC GCIA Version 3.3 Practical Detect - Rich Baker

Do ACK Floods normally contain payload? And have the push flag set? Could this be a response to stimulus? Can we learn anything about the destination machine? What were the parameters used when outputting what I perceive to be tcpdump output?
I'm not presuming to know the answers, but these are questions that come up when looking at your analysis.
Gary Morris

My response:

1. Do ACK Floods normally contain payload and have the push flag set?

Response: My research into the subject revealed that the attack may or may not contain a payload (the larger the more space it takes up in the host's connections table). As for the push flag, it makes no difference if the flag is set or not.

2. Could this be a response to stimulus?

It sure looks like it could to me, especially noting that the size of the payload varies. I would think that if you were taking the time to build one of these and adding a payload, it most likely wouldn't vary. Still, I guess it could if one so chose. If it wasn't for the fact that this attack was coming out of an IANA Reserved IP I'd probably not given it a second thought. I think I'm missing something here. With that said, I feel confident in saying that if I worked for the destination host I'd know if this type of transaction looked normal or not and if it was deserving of my attention or not.

3. Can we learn anything about the destination machine?

Response: As this was a port 80 attack, I tried the URL and it took me to "HITBOX Gateway." All I found was a blank page with the words Hitbox Gateway. A Whois query yielded the following results.

ARIN Whois search results for: 64.154.80.51

OrgName: Level 3 Communications, Inc.
OrgID: LVLT
Address: 1450 Infinite Drive Louisville CO 80027
Country: US
NetRange: 64.152.0.0 - 64.159.255.255
CIDR: 64.152.0.0/13
NetName: LC-ORG-ARIN
NetHandle: NET-64-152-0-0-1
Parent: NET-64-0-0-0-0

NetType: Direct Allocation
NameServer: NS1.LEVEL3.NET
NameServer: NS2.LEVEL3.NET
Comment: ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE
RegDate: 2000-06-08
Updated: 2001-05-30
Updated: 2001-05-30

The Level 3 Communications, Inc. (a very large communications Corporation) Web site can be viewed at: http://www.level3.com.

4. What were the parameters used when outputting what I perceive to be tcpdump output?

I just used the standard commands: ./tcpdump –r raw data_file_name > raw data_ file_name.txt

Summary:

As stated above this appears to be a weak attempt at an ACK flood. I can't come up with another reason why this data, coming from an IANA reserved IP, would continue for such a long period with no response back from the destination IP.

As IANA has the entire 46 IP range reserved, I thought I would run an ARIN Whois for just the first octet "046" and got the following results.

Search results for: 46
OrgName: Rutgers University
OrgID: RUTGER
ASNumber: 46
ASName: RUTGERS
ASHandle: AS46
Comment:
RegDate: 1985-08-16
Updated: 2000-08-10
TechHandle: RU-ORG-ARIN
TechName: Rutgers University Computing Services
TechPhone: +1-732-445-0327
TechEmail: netmanager@tdmx.rutgers.edu

If (and I do mean if) the 046 IP range is reserved for Rutgers University, than as I said at the start of this detect analysis, perhaps this is indeed a beginner (college student) testing the waters while hiding behind an IANA reserved IP. Of course, if this IP does belong to Rutgers University the IP would most likely not be spoofed.

Now, with that said, there is lots about this data that makes me think I could be misinterpreting it. Such facts as the destination host being Level 3 Communications, Inc., such a company would be expected to have volumes of data flowing through it. Another is the fact that the data load being pushed changes, causing one to think that maybe this is a response to stimulus (but where is the stimulus? It must be coming from a different IP.). And finally, the fact that I couldn't correlate this IP with any other reported attacks. This is where it would be nice to actually work at Level 3 Communications and have knowledge of the data that flows through it and the customers you service.

Detect 2: FTP port 21 unauthorized log-on attempt

Attempt 1.
217.85.152.57|(pD9559839.dip.t-dialin.net)||DoD.site.1|(Gov.site1.mil)
||TCP|4283|21|ftp|abc1|020925.10|09:22:48|0:0:3|3|39|3|45|

Attempt 2.
217.85.152.57|(pD9559839.dip.t-dialin.net)||DoD.site.2|(Gov.site2.mil)
||TCP|3922|21|ftp|def1|020925.10|09:55:25|0:0:4|5|39|5|124|

Attempt 3.
217.85.152.57|(pD9559839.dip.t-dialin.net)|| DoD.site.3|(Gov.site3.mil)
||TCP|4667|21|ftp|ghi1|020925.11|10:12:03|0:0:3|6|39|6|237|

Source of Trace:

This FTP port 21 attempted log-on was captured from my place of employment, a Department of Defense site.

Detect was generated by:

A government network intrusion detection system (IDS) referred to as Joint Intrusion Detection (JID) using software version 2.3.1 set to alert on the foot print "user anonymous" and displayed using a front-end browser developed by the Department of Defense (DoD) called a Joint Analysis Browser (JAB).

Probability the source address was spoofed:

Low. The attacker must establish a connection (TCP 3-way handshake) and receive log-on permission in order to gain access.

A resolution of the source address revealed the following:

inetnum: 217.80.0.0 - 217.89.31.255
netname: DTAG-DIAL14
descr: Deutsche Telekom AG
country: DE
admin-c: DTIP-RIPE
tech-c: ST5359-RIPE
status: ASSIGNED PA
changed: auftrag@nic.telekom.de 20020108
source: RIPE

Description of attack:

The attack above shows the source IP connecting to ftp port 21 of three Department of Defense sites. Closer examination of the data reveals the source sent 39 bytes of data to the destination. This is enough data for an anonymous log on. Streaming the data revealed that an anonymous log on was attempted using slightly varying passwords. The attempts were not successful.

Attempt 1.
==========
Source = 217.85.152.57 -- (Unknown)
Destination = DoD.site.1 -- (Unknown)
Start time = Wed Sep 25 09:22:48 2002
Protocols = [4283 21] (6)
Stream = connections.log.5.stream.init
==========
USER anonymous
PASS Ggpuser@home.com
[***** End of stream *****]
==========
Source = DoD.site.1 -- (Unknown)
Destination = 217.85.152.57 -- (Unknown)
Start time = Wed Sep 25 09:22:48 2002
Protocols = [21 4283] (6)
Stream = connections.log.5.stream.dest
530 Connection refused, unknown IP address.
[***** End of stream *****]

Attempt 2.
==========

```
Source      =  217.85.152.57 -- (Unknown)
Destination =    Dod.site.2 -- (Unknown)
Start time  = Wed Sep 25 09:55:25 2002
Protocols   = [3922 21] (6)
Stream      = connections.log.51.stream.init
================================
USER anonymous
PASS Sgpuser@home.com
  [***** End of stream *****]
================================
Source      =    Dod.site.2 -- (Unknown)
Destination =  217.85.152.57 -- (Unknown)
Start time  = Wed Sep 25 09:55:25 2002
Protocols   = [21 3922] (6)
Stream      = connections.log.51.stream.dest
================================
220 rsu1_col Microsoft FTP Service (Version 5.0).
331 Password required for anonymous.
530 User anonymous cannot log in.
  [***** End of stream *****]

Attempt 3.
================================
Source      =  217.85.152.57 -- (Unknown)
Destination = Dod.site.3 -- (Unknown)
Start time  = Wed Sep 25 10:12:03 2002
Protocols   = [4667 21] (6)
Stream      = connections.log.14.stream.init
================================
 USER anonymous
PASS Dgpuser@home.com
  [***** End of stream *****]
================================
Source      = Dod.site.3 -- (Unknown)
Destination =  217.85.152.57 -- (Unknown)
Start time  = Wed Sep 10:12:03 2002
Protocols   = [21 4667] (6)
Stream      = connections.log.14.stream.dest
================================
220-FTPD1 IBM FTP CS V2R10 at MSS-FTP.XYZ.1234.MIL, 10:09:44 on 2002-
09-25.
220 Connection will close if idle for more than 5 minutes.
331 Send password please.
530 PASS command failed - __passwd() error : EDC5143I No such process.
   [***** End of stream *****]
```

Attack mechanism:

The attacker IP must first scan for FTP servers. Once found, the source IP logs on and attempts to create a directory. In order for the compromised server to be able to send data to the attacker, the attacker must open a second connection between it and the server. To accomplish this the attacker sends a port command to the server identifying the IP address and port it desires to use for data transfer. With this information the compromised server opens the connection via its port 20 allowing the attacker to transfer whatever data it desires to the compromised server.

Correlations:

A search of DShield.Org for IP 217.85.152.57 yielded no results. A search of our own DoD IP data base revealed this IP had performed an ftp port 21 scan of various DoD IP ranges which included those referenced above. As a result of the original ftp scan by IP 217.85.152.57 a report was forwarded to the scanned locations recommending they block this IP.

References to FTP scans, anonymous logon, vulnerabilities, etc abound and several can be found at www.cert.org. An excellent paper on FTP analysis can be found at http://www.eyeonsecurity.net/papers/ftpscanning.html.

Evidence of activity targeting:

Moderate. While it is possible that this was an act of random anonymous logon attempts to public FTP servers. It's important to note that the source IP resolved to Germany, variants of the same password was used at each location and the IPs attacked were of the same government agency.

Severity:

Criticality = 5 (while it is uncertain if the attack was targeted, a FTP log on to these government sites has potential for server consequences and should be treated accordingly)

Lethality = 5 (if successful the attacker could cause server consequences)

System Countermeasures = 4 (this is a modern operating system, all patches, added security such as tcp wrappers and secure shell, etc…but there is no such thing as the perfectly defended system.)

Network Countermeasures = 4 (a validated restrictive firewall is in use, only one way in or out… but there is no such thing as the perfectly defended network.)

Severity = [(5+5) – (4+4)] = 2

Defensive recommendation:

Blocking this IP address is a reasonable course of action. Additionally, ensure anonymous logon is not allowed and that all authorized logons require username/password authentication. It is further recommended that only downloading of files be allowed. This action will protect against denial of service attacks in addition to preventing the uploading of malicious programs and files. If uploads are permitted, it is recommended they be allowed only to a specified folder on a separate physical drive from the operating system and monitored regularly.

Multiple choice test question:

What action provides the greatest protection against unauthorized FTP logon?

A. Don't allow ftp anonymous logon
B. Allow only authorized ftp logon using username/password authentication
C. Block all ftp traffic
D. Both A and B

Correct response: C

Detect 3: Code Red Worm

To save space "- - -" indicates data not posted.

```
NIDabc1 [2002-09-09 15:23:28] [arachNIDS/552] [CVE/CAN-2000-0071] [snortDB/1243]  WEB-
IIS ISAPI .ida attempt
IPv4: 12.238.60.15 -> DoD.Site.gov
     hlen=5 TOS=16 dlen=2964 ID= flags= offset= TTL=240 chksum=
TCP:  port=4005 -> dport: 80  flags=***AP*** seq=2440586701
     ack=2306511251 off=5 res= win=8760 urp= chksum=0
Payload:  length = 2809

000 : 47 45 54 20 2F 64 65 66 61 75 6C 74 2E 69 64 61   GET /default.ida
010 : 3F 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E   ?NNNNNNNNNNNNNNN
020 : 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E   NNNNNNNNNNNNNNNN
- - -
0e0 : 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E 4E   NNNNNNNNNNNNNNNN
0f0 : 4E 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00   N...............
100 : C3 03 00 00 00 78 00 FA 20 25 75 39 30 39 30 25   .....x.. %u9090%
110 : 75 36 38 35 38 25 75 63 62 64 33 25 75 37 38 30   u6858%ucbd3%u780
120 : 31 25 75 39 30 39 30 25 75 36 38 35 38 25 75 63   1%u9090%u6858%uc
```

```
- - -
370 : 72 22 3E 57 65 6C 63 6F 6D 65 20 74 6F 20 68 74   r">Welcome to ht
380 : 74 70 3A 2F 2F 77 77 77 2E 77 6F 72 6D 2E 63 6F   tp://www.worm.co
390 : 6D 20 21 3C 62 72 3E 3C 62 72 3E 48 61 63 6B 65   m !<br><br>Hacke
3a0 : 64 20 42 79 20 43 68 69 6E 65 73 65 21 3C 2F 66   d By Chinese!</f
```

Source of Trace:

This Code Red worm was captured from my place of employment, a Department of Defense site.

Detect was generated by:

SNORT version 1.86 using customized rule set and displayed using ACID version 0.9.6b22.

Probability the source address was spoofed:

Low. As the Code Red worm is designed to spread itself from compromised host to compromised host, it is very unlikely the source address is spoofed.

A resolution of the source address revealed the following:

Search results for: 12.238.60.15
OrgName: AT&T WorldNet Services
OrgID: ATTW

NetRange: 12.0.0.0 - 12.255.255.255
CIDR: 12.0.0.0/8
NetName: ATT
NetHandle: NET-12-0-0-0-1
Parent:
NetType: Direct Allocation
NameServer: DBRU.BR.NS.ELS-GMS.ATT.NET
NameServer: DMTU.MT.NS.ELS-GMS.ATT.NET
NameServer: CBRU.BR.NS.ELS-GMS.ATT.NET
NameServer: CMTU.MT.NS.ELS-GMS.ATT.NET
Comment: For abuse issues contact abuse@att.net
RegDate: 1983-08-23
Updated: 2002-08-23

Description of attack:

The above attack shows the source IP attempting to connect to port 80. The attempt is successfully blocked and access is denied.

Code Red activity can be identified by the following signature/footprint or a variant, but always contains the /default.ida? signature:

/default.ida?
NNN%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801%u9090%u6858%ucbd3%u7801%u9090%u9090%u81

Attack mechanism:

Referencing CERT Advisory CA-2001-19, Code Red worm, is a self-replicating malicious code designed to take advantage of a "buffer overflow" bug in Microsoft's Internet Information Services (IIS) Web server software. It does this by attempting to connect to TCP port 80 of randomly selected hosts in hopes of finding a Microsoft Web server. If the attacking host is successful in connecting to port 80 it sends a crafted HTTP GET request to its victim designed to exploit the IIS buffer overflow bug.

Due to Code Red's self-replicating nature it sends the same crafted HTTP GET request to each of its randomly selected hosts. Each of these hosts may react differently to the crafted request based on its individual configuration.

Examples below were taken from http://www.cert.org/advisories/CA-2001-19.html.

- IIS 4.0 and 5.0 servers with indexing service installed (not patched) will almost certainly be compromised by the Code Red worm.
- Unpatched Cisco 600-series DSL routers will process the HTTP request thereby triggering an unrelated vulnerability which causes the router to stop forwarding packets. (http://www.cisco.com/warp/public/707/cisco-code-red-worm-pub.shtmi)
- Systems not running IIS, but with an HTTP server listening on TCP port 80 will probably accept the HTTP request, return with an "HTTP 400 Bad Request" message, and potentially log this request in an access log.

The early version of Code Red defaced English language default web pages requested by the server with the statement "HELLO! Welcome to http:/www.worm.com! Hacked by the Chinese!"

Code Red also produces time sensitive activity based on the day of the month using the system clock. (http://www.cert.org/advisories/CA-2001-19.html)

- Day 1 – 19: self-replicate by randomly selecting IP addresses and attempting to connect to TCP port 80.
- Day 20 – 27: Initiate a packet-flooding denial of service attack against a specific target.

- Day 28 through remainder of month: No activity.

It should be noted that later variants of Code Red did more than just deface web pages, initiate packet-flooding denial of service, and seek to replicate itself. Using the IIS vulnerability, variants of Code Red can leave backdoors and allow an attacker to execute arbitrary code in the local system security context allowing the attacker to take control of the compromised system.

Correlations:

A search of DShield.Org for IP 217.85.152.57 yielded the following results.

Rows returned: 236
Result:
date|time (gmt)|count|sourceip|sourceport|targetport|protocol|flags
2002-09-09|00:02:39|1|012.238.060.015|3023|80|6|
2002-09-09|00:02:42|1|012.238.060.015|3023|80|6|
2002-09-09|00:02:48|1|012.238.060.015|3023|80|6|
2002-09-09|00:26:17|1|012.238.060.015|3620|80|6|S
….
….
….
2002-09-09|23:33:44|1|012.238.060.015|3868|80|6|
2002-09-09|23:59:40|1|012.238.060.015|4663|80|6|S
2002-09-09|23:59:43|1|012.238.060.015|4663|80|6|S
2002-09-09|23:59:49|1|012.238.060.015|4663|80|6|S

While it's not certain that the IP visited port 80 on this date for the purpose of spreading Code Red worm, it can be assumed that since it brought the worm to our IP on this date it most likely did the same to these.

Code Red worm is well documented and information abounds at http://www.cert.org.

Evidence of activity targeting:

Moderate. The host does try to replicate itself by actively targeting TCP port 80 IIS Microsoft servers, but does this via randomly selected IP addresses. So, if Code Red worm attacks a specific server, the odds are it is more by chance than design.

Severity:

Criticality = 4 (while Code Red worm selects IP addresses randomly, it does target IIS Microsoft servers and its capabilities are well documented)

Lethality = 5 (Code Red can do more than just deface web pages, initiate

packet-flooding denial of service, and seek to replicate itself. Using the IIS vulnerability, variants of Code Red can leave backdoors and allow an attacker to execute arbitrary code in the local system security context allowing the attacker to take control of the compromised system.)

System Countermeasures = 4 (this is a modern operating system, all patches, added security such as tcp wrappers and secure shell, etc…but there is no such thing as the perfectly defended system.)

Network Countermeasures = 4 (a validated restrictive firewall is in use, only one way in or out…but there is no such thing as the perfectly defended network.)

Severity = [(4+5) – (4+4)] = 1

Defensive recommendation:

Blocking this IP address is a reasonable course of action. Ensure IIS Microsoft servers are properly patched against all known vulnerabilities and stay abreast of current updates. Get and use anti-virus software and regularly updating anti-virus software will protect the system from new viruses and worms.

Multiple choice test question:

What is the best defense against Code Red worm?

A. Become an educated IT user
B. Update and use anti-virus software
C. Ensure IIS Microsoft servers are properly patched
D. All of the above

Correct response: D

PART – 3 ANALYZE THIS (40 points)
You have been asked to provide a security audit for a University by analyzing logs from their intrusion detection system collected over a five day period**.**

Note for GCIA grader: I contacted SANS to find out what course of action I should take as none of the data available for analysis met the "five consecutive day, not older than 90 day" criteria. I received permission from Jamie French, GCIA Lead Grader, SANS Institute to use the June data below for this analysis.

An analysis of University log files captured by Snort beginning 12 June and ending 17 June 2002 is provided below. This analysis was to encompass five consecutive day, but data for 16 June was not available/complete. To ensure five days of data was evaluated, data from 17 June was included in the evaluation.

University log files analyzed are as follows:

Snort Alerts from files; alert.0206[12-15 & 17]
Scan Detections from files; scansscans.0206[12-15 & 17].clean
Out of Spec packets from files; oos.Jun.[12-15 & 17].2002

EXECUTIVE ANALYSIS SUMMARY

An analysis of the University logs revealed 222,686 Snort alerts (not to include port scans), 443,230 scan detections, and 26 out-of-spec packets.

* The top 10 Snort alerts identified are as follows:
1.) SMB Name Wildcard
2.) SNMP public access
3.) spp_http_decode IIS Unicode attack detected
4.) INFO Possible IRC Access
5.) ICMP Echo Request L3retriever Ping
6.) MISC Large UDP Packet
7.) INFO MSN IM Chat data
8.) AFS - Off-campus activity
9.) NFO Inbound GNUTella Connect request
10.) suspicious host traffic

* The top 10 scan detections are as follows:
1.) UDP
2.) SYN
3.) INVALIDACK
4.) NOACK
5.) NULL
6.) UNKNOWN
7.) VECNA
8.) FIN
9.) NMAPID
10.) FULLXMAS

* Out-of Spec packet analysis revealed multiple TCP flag set irregularities.

Log analysis suggests a lack of network security. A complete review of current security policy is recommended (to include university policy on internet gaming, audio/visual downloads/streaming, etc.). Further, it is suspected that a worm/virus such as Nimda has compromised the network. As a result of this analysis it is recommended that the network by taken off-line until corrective action is taken, proper safeguards are in place, and a new comprehensive security policy is established and functioning.

Furthermore, after corrective actions have been taken and the network is returned to on-line operation, a follow-up logs analysis is recommended to ensure the network is operating properly and in accordance with the new security policy.

A more detailed analysis of Snort alerts, scan detections, and out-of-spec packets of the June logs follows.

ALERT ANALYSIS

An evaluation of alert data yielded the following results.

46947 SMB Name Wildcard
45874 SNMP public access
38965 spp_http_decode IIS Unicode attack detected
21958 INFO Possible IRC Access
21792 ICMP Echo Request L3retriever Ping
11162 MISC Large UDP Packet
8240 INFO MSN IM Chat data
5119 AFS - Off-campus activity
3678 INFO Inbound GNUTella Connect request
2994 suspicious host traffic
2891 ICMP Echo Request Nmap or HPING2
2497 High port 65535 udp - possible Red Worm - traffic

2368 spp_http_decode CGI Null Byte attack detected
1187 ICMP Fragment Reassembly Time Exceeded
1169 WEB-MISC Attempt to execute cmd
933 ICMP Router Selection
609 FTP DoS ftpd globbing
540 WEB-IIS view source via translate header
525 ICMP Destination Unreachable
522 Incomplete Packet Fragments Discarded
392 INFO Outbound GNUTella Connect request
336 Null scan!
242 SCAN Proxy attempt
215 NIMDA - Attempt to execute cmd
215 IDS552/web-iis_IIS ISAPI Overflow ida nosize
192 ICMP Echo Request Windows
101 ICMP Echo Request CyberKit 2.2 Windows
99 INFO - Possible Squid Scan
93 Watchlist 000220 IL-ISDNNET-990517
80 IRC evil - running XDCC
72 ICMP traceroute
70 INFO FTP anonymous FTP
66 SUNRPC highport access!
62 WEB-FRONTPAGE _vti_rpc access
61 WEB-IIS _vti_inf access
52 ICMP Echo Request BSDtype
46 Possible trojan server activity
34 WEB-MISC http directory traversal
28 WEB-MISC 403 Forbidden
26 INFO Napster Client Data
23 NMAP TCP ping!
22 Attempted Sun RPC high port access
19 UDP SRC and DST outside network
15 WEB-CGI scriptalias access
14 Watchlist 000222 NET-NCFC
11 WEB-MISC compaq nsight directory traversal
11 SCAN FIN
10 WEB-IIS Unauthorized IP Access Attempt
10 Queso fingerprint
9 SCAN Synscan Portscan ID 19104
9 EXPLOIT x86 NOOP
8 INFO Inbound GNUTella Connect accept
8 Back Orifice
6 MISC traceroute
6 EXPLOIT x86 setuid 0
6 EXPLOIT x86 setgid 0
6 EXPLOIT NTPDX buffer overflow
4 High port 65535 tcp - possible Red Worm - traffic
3 WEB-CGI redirect access
3 Port 55850 tcp - Possible myserver activity - ref. 010313-1

3 MISC source port 53 to <1024
3 MISC PCAnywhere Startup
2 Virus - Possible scr Worm
2 Virus - Possible pif Worm
2 Port 55850 udp - Possible myserver activity - ref. 010313-1
1 X11 outgoing
1 WEB-CGI formmail access
1 Virus - Possible MyRomeo Worm
1 TFTP - Internal UDP connection to external tftp server
1 TFTP - External UDP connection to internal tftp server
1 SCAN XMAS
1 Probable NMAP fingerprint attempt
1 INFO Outbound GNUTella Connect accept
1 EXPLOIT x86 stealth noop

The alert file contained 222,686 Snort Alerts (not to include port scans) and 74 unique alert types. It is not necessary to analyze each and ever one of these alerts to gain an overall understanding of the basic network security posture. As the top 10 alerts account for 93% (206,729 alerts) of the logged alerts (not to include port scans) our time would be best spent evaluating these main targets. By doing our analysis in this manner our efforts are better focused and our time is better utilized while insuring over 90% of the detected alerts are evaluated.

A complete analysis involves more than examining alerts. It is important that we understand that the most common alerts are not necessarily the most serious threats. So, with that in mind, after evaluating these alerts we will examine Scan data and Out-of-Spec packets.

The following is a list of the top 10 source and destination hosts (Top-Talkers) taken from the University alerts logs (not to include port scans). Selection criteria is based on the number of alerts associated with the host. Each of the top 10 alerts will be addressed below individually.

Top 10 Alert Source Hosts:

21,995	-	130.85.151.90 (host)
11,161	-	130.85.11.7
9,483	-	130.85.70.177
9,307	-	130.85.11.6
8,239	-	130.85.88.181
7,621	-	130.85.153.179
5,813	-	202.102.249.118
4,674	-	130.85.153.136
4,154	-	130.85.88.203
4,114	-	130.85.88.159

Top 10 Alert Destination Hosts:

25,544	-	130.85.150.195 (host)
23,620	-	130.85.11.7
19,624	-	130.85.11.6
7,802	-	66.28.132.168
7,516	-	66.62.70.248
6,657	-	64.246.34.181
5,876	-	130.85.88.140
4,533	-	130.85.150.84
3,809	-	130.85.153.142
3,323	-	130.85.5.96

Alert 1: SMB Name Wildcard

Top 5 Source IPs

11,161	-	130.85.11.7 (host)
9,307	-	130.85.11.6
2,973	-	130.85.5.89
1,021	-	130.85.11.5
997	-	130.85.152.172

Top 5 Destination IPs

10,992	-	130.85.11.7 (host)
9,187	-	130.85.11.6
1,011	-	130.85.152.172
996	-	130.85.11.5
486	-	130.85.152.186

This is a standard netbios name table retrieval query. Windows machines often exchange these queries as a part of the file sharing protocol to determine NetBIOS names when only IP addresses are known. An attacker could use this same query to extract useful information such as workstation name, domain, and users currently logged in. There are two malicious sources for this alert, they are: 1.) intruders using Netbios can discover information about the network, and 2.) a worm known as network.vbs.

Windows machines typically send these types of queries in normal operation, particularly when file sharing is active, to determine NetBIOS names when only IP addresses are known. This type of query, when originating from an external network, is usually a pre-attack probe to gather netbios name table information such as workstation name, domain, and a list of currently logged in users. This signature was created and can be reproduced
by using the unix samba command "nmblookup -A ". By accessing system name table information, individuals can obtain information which can be used to launch an attack. Information available includes: 1.) The NetBIOS name of the server. 2.) The Windows NT workgroup domain name. 3.) Login names of users who are logged into the server. 4.) The name of the administrator account if they are logged into the server.

It is considered best practice to ensure that users outside of your network are not permitted to access the NetBIOS name service. This is usually accomplished by configuring packet filters to drop UDP traffic to port 137.

Source: http://archives.neohapsis.com/archives/snort/2000-01/0222.html
http://www.sans.org/newlook/resources/IDFAQ/port_137.htm

Alert 2: SNMP Public Access

Top 5 Source IPs
9,467 - 130.85.70.177 (host)
8,233 - 130.85.88.181
4,154 - 130.85.88.203
4,114 - 130.85.88.159
4,088 - 130.85.88.207

Top 5 Destination IPs
25,494 - 130.85.150.195 (host)
4,487 - 130.85.150.84
2,201 - 130.85.5.97
2,154 - 130.85.5.127
2,153 - 130.85.5.96

SNMP agents provide a huge amount of sensitive information. The protection of SNMP information is usually based on a community string which should be kept secret (but is transmitted in the clear in some versions of SNMP). If SNMP agents are initially configured they often come with a default community string set by the vendor. If this default is not changed an attacker that tries known default will get the same access as the authorized users. The same results will be achieved if an attacker can guess the correct (changed) password. The information available to an attacker might consist of: - running services - shares - users and domains.

CERT Advisory CA-2002-03 “Multiple Vulnerabilities in Many Implementations of the Simple Network Management Protocol (SNMP)” dated September 23, 2002 offers several solutions these vulunerabilities, but warns these solutions may have significant impact on your everyday network operations and/or network architecture.

Source: http://ki.sei.cmu.edu/idar/drill_attack.cfm?attack=SNMP%20Grabbing
http://www.cert.org/advisories/CA-2002-03.html

Alert 3: spp_http_decode IIS Unicode attack detected

Top 5 Source IPs
7,620 - 130.85.153.179 (host)
4,674 - 130.85.153.136
3,564 - 130.85.88.201
2,134 - 130.85.153.163
1,958 - 130.85.153.195

Top 5 Destination IPs
3,300 - 211.210.13.212 (host)
2,295 - 211.239.164.163
1,880 - 211.63.185.30
1,500 - 211.239.164.180
1,231 - 211.115.213.202

Basically, it looks like we have a real problem here. The top 5 source IPs for this alert belong to MY.NET (130.85.XXX.XXX, the University). These machines are attempting a directory traversal attack on other IIS hosts, possibly using the unicodexecute2.pl perl script. This alert is generally caused by Nimda. A closer examination of these machines are in order.

Alert: spp_http_decode: IIS Unicode attack detected
Alert: High port 65535 udp – possible Red Worm - traffic
Alert: WEB-MISC Attempt to execute

The above three alerts are all well known signatures of Code Red worm and the Nimda worm/virus infections of Microsoft's IIS web server. Once infected, the host will scan port 80 looking for other Microsoft IIS web servers to infect. Combined, these three alerts make up 42,631 alerts, that's 19% of the alert detections on this system. What all this adds up to is the importance of ensuring all current Microsoft patches are in place, anti-virus software is current and both are updated regularly. Finally, it's important to keep an eye on your alerts checking for possible infection.

Source: http://www.winplanet.com/winplanet/tips/3787/1
http://www.f-secure.com/v-descs/nimda.shtml

Alert 4: INFO Possible IRC

Top 5 Source IPs
21,896 - 130.85.151.90 (host)
9 - 130.85.71.236
9 - 130.85.116.69
8 - 130.85.150.165
7 - 130.85.163.93

Top 5 Destination IPs

7,788	-	66.28.132.168 (host)
7,458	-	66.62.70.248
6,650	-	64.246.34.181
16	-	64.246.44.97
13	-	207.68.167.253

There Are several kinds of Internet chat. The most common types of chat are IRC, Webpage (Java) Chat, and Instant Messenger Chat. Internet Relay Chat (IRC) is one of the most popular and most interactive services on the Internet. Mirc.Com compares IRC to the Citizen's Band radio, saying "IRC is the net's equivalent of CB radio. But unlike CB, Internet Relay Chat lets people all over the world participate in real-time conversations." IRC is the most widely used means of chatting with people around the world. According to newircusers.com, IRC is worldwide and at any given time more than 100,000 people are online chatting, 24 hours a day.

Internet Relay Chat is part of today's university education experience, so it isn't going away. This is evidenced by the fact that the source IP 130.85.151.90 for this alert belongs to the university. However, from a security standpoint, routers, firewalls and other security safeguards should be reviewed and to ensure maximum protection.

ARIN Whois search results for: 130.85.151.90

OrgName: University of Maryland Baltimore County
OrgID: UMBC
Address: UMBC University Computing Baltimore MD 21250
Country: US
NetRange: 130.85.0.0 - 130.85.255.255
CIDR: 130.85.0.0/16
NetName: UMBCNET
NetHandle: NET-130-85-0-0-1
Parent: NET-130-0-0-0-0
NetType: Direct Assignment
NameServer: UMBC5.UMBC.EDU
NameServer: UMBC4.UMBC.EDU
NameServer: UMBC3.UMBC.EDU
Comment:
RegDate: 1988-07-05
Updated: 2000-03-17

Source: http://www.mirc.com/irc.html
http://www.newircusers.com/java/index.html
http://www.newircusers.com/java/security.html

Alert 5: ICMP Echo Request L3retriever Ping

Top 5 Source IPs

1,001	-	130.85.152.172 (host)
492	-	130.85.152.186
479	-	130.85.152.161
461	-	130.85.152.171
461	-	130.85.152.166

Top 5 Destination IPs

11,030	-	130.85.11.7 (host)
9,212	-	130.85.11.6
1,023	-	130.85.11.5
246	-	130.85.5.4
226	-	130.85.104.177

This alert is caused by either the L3 “Retriever 1.5” security scanner, or by a Windows 2000 host. An interesting correlation identified by Joe Ellis in his GCIA practical between the ICMP Echo Request L3retriever Ping alert and SMB Name Wildcard was also revealed in the data used for this analysis. It was noted that the top destination host for this alert was also the top source and destination host in the SMB Name Wildcard alert. The type of pattern seen below is seen repeatedly in the alert logs:

06/15-01:33:17.214871 [**] ICMP Echo Request L3retriever Ping [**] 130.85.152.248 -> 130.85.11.7
06/15-01:33:17.211427 [**] SMB Name Wildcard [**] 130.85.152.248 -> 130.85.11.7
06/15-01:33:17.215671 [**] SMB Name Wildcard [**] 130.85.11.7:137 -> 130.85.152.248:137

As Joe Ellis points out in his paper, the above pattern would indicate IP 130.85.152.248 is a Windows 2000 host and it is querying 130.85.11.7 for Netbios domain information.

Source:http://www.whitehats.com/cgi/arachNIDS/show?_id=ids311&view=event
http://www.giac.org/practical/Joe_Ellis_GCIA.doc

Alert 6: MISC Large UDP Packet

Top 5 Source IPs

5,810	-	202.102.249.118 (host)
1,642	-	211.63.185.15
1,107	-	211.63.185.21
815	-	208.252.239.125
716	-	202.210.163.74

Top 5 Destination IPs

5,810	-	130.85.88.140 (host)
2,749	-	130.85.153.179
716	-	130.85.152.21
633	-	130.85.150.209
552	-	130.85.153.115

This alerts goes off when an abnormally large UDP packet is detected by the server and could indicate a denial of service attack. How the Snort rule is set will determine how large a UDP packet is required to set off this alert. This is normally a packet over 4000 bytes. UDP is excellent for transporting large amounts of data, such as Internet games, where data integrity is not necessary.

A review of the top 5 Misc Large UDP Packet source IPs shows that the top source IP 202.102.249.118 is also found in the overall top 10 source IPs. It should also be noted that the top MISC Large UDP Packet destination IP 130.85.88.140 is in the overall top 10 destination IPs. Knowing more about these large UDP packet source IPs could give us insight into how they came to our network. An examination of the top 5 source IPs for this alert revealed the following:

202.102.249.118
Asia Pacific Network Information Center (http://www.apnic.net)
inetnum: 202.102.249.0 - 202.102.249.255
netname: ZZTB-MIB
descr: Zhengzhou Telecom bureau Multimedia Information Bureau,
descr: Zhengzhou city, Henan Provice
descr: 450052
country: CN
admin-c: LZ33-AP
tech-c: LZ33-AP
mnt-by: MAINT-CHINANET-HA
changed: zhail@email.online.ha.cn 20010302
status: ALLOCATED PORTABLE
source: APNIC

This IP address is an audio/visual link from China. An examination of the IP revealed the following:

202.102.249.118
|___ 21 File Transfer Protocol [Control]
|___ 220 Serv-U FTP Server v3.0 for WinSock ready.....
|___ 53 Domain Name Server
|___ 80 World Wide Web HTTP
|___ HTTP/1.1 404 Object Not Found..Server: Microsoft-IIS/5.0..Date: Mon, 04 Nov 2002 19:00:39 GMT..Content-Type: text/html..Content
|___ 135 DCE endpoint resolution
|___ 139 NETBIOS Session Service
|___ 445 Microsoft-DS

|___ 1025 network blackjack
|___ 1027 ICQ?
|___ 1433 Microsoft-SQL-Server
|___ 1755 ms-streaming
|___ 7007 basic overseer process

Further research into source IP 202.102.249.118 showed that every Internet transaction that it had with the University was to destination MY.NET.88.140 and each transaction (5,810) resulted in a MISC Large UDP Packet alert. It would be worth while checking with the users of MY.NET.88.140 to determine if they are aware of these transactions, if they were for multimedia streaming traffic, and if not, what was the purpose. The IP may require blocking. Bottom line…further analysis of this machine is in order.

211.63.185.15 and 211.63.185.21

Korea Network Information Center (http://whois.nic.or.kr/english/index.html)
IP Address : 211.63.185.0-211.63.185.255
Network Name : KORNET-IDC-JUNGANG-KTIDC
Connect ISP Name : KORNET
Connect Date : 20000417
Registration Date : 20010308
[Organization Information]
Orgnization ID : ORG203787
Org Name : CENTRAL DATA COMMUNICATION OFFICE
State : SEOUL
Address : 128-9 YEUNKEONDONG JONGROKU
Zip Code : 110-460
[Admin Contact Information]
Name : GilSoon Park
Org Name : KOREA TELECOM
State : SEOUL
Address : 128-9 Youngundong Chongroku
Zip Code : 110-460
Phone : +82-2-747-9213
Fax : +82-2-766-5901
E-Mail : gspark@kornet.net

These two IP addresses are from South Korea. An examination of both IPs revealed the following:

211.63.185.15
|___ 1755 ms-streaming
|___ 8080 Standard HTTP Proxy

211.63.185.21
|___ 1755 ms-streaming
|___ 8080 Standard HTTP Proxy

These MISC Large UDP Packet alerts appear to be multimedia streaming traffic.

208.252.239.125
ARIN Whois Data Search (http://www.arin.net/whois/)
OrgName: THQ, Inc.
OrgID: THQINC
Address: 27001 Agoura Rd Calabasas Hills CA 91301
Country: US
NetRange: 208.252.239.0 - 208.252.239.127
CIDR: 208.252.239.0/25
NetName: UU-208-252-239
NetHandle: NET-208-252-239-0-1
Parent: NET-208-192-0-0-1
NetType: Reassigned
Comment:
RegDate: 2001-06-19
Updated: 2002-05-24
TechHandle: RR1389-ARIN
TechName: Riley, Rob
TechPhone: +1-818-871-5028
TechEmail: rriley@thq.com

This IP address is from CA, USA. None of my attempts to learn more about this IP (a non-active host) succeeded. It's possible this machine is disconnected or just turned off and is a catch-as-catch-can IP.

202.210.163.74
JPNIC Whois Gateway (http://whois.nic.ad.jp/cgi-bin/whois_gw)
Network Information:
a. [Network Number] 202.210.163.0
b. [Network Name] BEKKOAME-NET
g. [Organization] Network Service
m. [Administrative Contact] KO022JP
n. [Technical Contact] MK5026JP
p. [Nameserver] ns0.big.or.jp
p. [Nameserver] ns1.big.or.jp
y. [Reply Mail] big-horn@bekknet.ad.jp
y. [Reply Mail] tinkerbell@bekknet.ad.jp
[Assigned Date] 2000/04/20
[Return Date]
[Last Update] 2000/05/09 10:00:22 (JST)
hosting@bekknet.ad.jp

This IP address is an audio/visual link from Japan. An examination of the IP revealed the following:

202.210.163.74
|___ 21 File Transfer Protocol [Control]

|___ 220- Jgaa's Fan Club FTP Service WAR-FTPD 1.65 Ready..220 Please enter your user name...
|___ 80 World Wide Web HTTP
|___ HTTP/1.0 400 Bad Request..Server: Cougar 4.1.0.3927....
|___ 81 HOSTS2 Name Server
|___ 135 DCE endpoint resolution
|___ 139 NETBIOS Session Service
|___ 443 https MCom
|___ 445 Microsoft-DS
|___ 1025 network blackjack
|___ 1027 ICQ?
|___ 1755 ms-streaming
|___ 7007 basic overseer process

Tod A. Beardsley encountered this same type of activity and noted it in his May 8, 2002 GCIA analysis. His research of "Cougar 4.1.0.392…" server identified "Cougar" as a codename for NetShow. And, that "Windows Media Services" (nee Netshow nee Cougar) does, in fact, sometimes rely on large UDP packets to deliver its streaming content."

Source:http://www.whitehats.com/cgi/arachNIDS/show?_id=ids311&view=event
http://www.giac.org/practical/Joe_Ellis_GCIA.doc
http://www.giac.org/practical/Tod_Beardsley_GCIA.doc

Alert 7: INFO MSN IM Chat Data

Top 5 Source IPs

797	-	130.85.150.242 (host)
450	-	130.85.150.232
329	-	130.85.153.142
286	-	130.85.153.46
279	-	64.4.12.185

Top 6 Destination IPs

665	-	130.85.150.242 (host)
574	-	130.85.150.232
434	-	130.85.153.46
373	-	130.85.153.142
318	-	130.85.153.107
286	-	64.4.12.185

Microsoft Networks Instant Messenger (MSN IM) is a application which allows you to send instant messages, call anywhere in the world from your computer,

see when someone's typing, page a contact's mobile phone, send pictures and music to your friends, etc. The application connects to a central server, here seen as IP 64.4.12.185. This IP is number five (5) source and number six (6) destination of the INFO MSN IM Chat Data top source and destination IPs. A whois look up of this IP reveals the following.

Search results for: 64.4.12.185
OrgName: MS Hotmail
OrgID: MSHOTM

NetRange: 64.4.0.0 - 64.4.63.255
CIDR: 64.4.0.0/18
NetName: HOTMAIL
NetHandle: NET-64-4-0-0-1
Parent: NET-64-0-0-0-0
NetType: Direct Assignment
NameServer: NS1.HOTMAIL.COM
NameServer: NS3.HOTMAIL.COM
NameServer: NS2.HOTMAIL.COM
NameServer: NS4.HOTMAIL.COM
Comment:
RegDate: 1999-11-24
Updated: 2002-07-15

OrgName: MS Hotmail
OrgID: MSHOTM
Address: 1065 La Avenida Mountain View CA 94043
Country: US
Comment:
RegDate: 1999-11-24
Updated: 2002-07-15

There is a vulnerability for this application and it comes in the form of a worm. According to www.symantec, this worm uses the MSN Messenger Service (MSNMS) program to replicate itself. It does nothing more than replicate, and if it is executed on a computer that does not have MSNMS installed, it simply remains resident in memory without replicating.

Source:http://messenger.msn.com/
http://messenger.msn.com/support/features.asp?client=1
http://www.symantec.com/avcenter/venc/data/w32.choke.worm.html

Alert 8: AFS - Off-campus activity

Top 5 Source IPs
1,923 - 12.151.57.37 (host)
922 - 63.215.70.142

264 - 66.28.225.156
253 - 63.250.219.184
219 - 10.16.1.40

Top 5 Destination IPs
1,923 - 130.85.88.245 (host)
1,370 - 130.85.151.95
297 - 130.85.153.168
297 - 130.85.152.19
219 - 130.85.153.46

AFS is a distributed file system that joins the file systems of networked computers. AFS uses a client-server architecture, whereby special server computers deliver files and software to the client workstations that request them. AFS is a secure system requiring password authentication for users. AFS also requires mutual authentication between servers and clients whenever they communicate with each other.

GIAC Information Bulletin H-109: Solaris DCE and AFS Integrated login Vulnerability (http://www.ciac.org/ciac/bulletins/h-109.shtml) identifies the following vulnerability for Solaris systems.

PROBLEM: A vulnerability exists on systems running Transarc's Solaris DCE integrated login program (login.dce in place of /bin/login) which have AFS installed but no AFS klog binary in any of the standard locations.

PLATFORM: Solaris 2.4 and Solaris 2.5 running Transarc DCE 1.1 in conjunction with any version of AFS.

DAMAGE: Users without accounts on the system may gain unauthorized access to local resources.

SOLUTION: Apply patches or workaround as listed on http://www.ciac.org/ciac/bulletins/h-109.shtml.

Alert 9: INFO Inbound GNUTella Connect request

Top 5 Source IPs
12 - 217.81.69.230 (host)
5 - 212.244.197.133
5 - 12.162.224.20
4 - 68.45.248.244
4 - 68.39.181.17

Top 5 Destination IPs
3,329 - 130.85.153.142 (host)
249 - 130.85.150.209

41 - 130.85.100.157
30 - 130.85.88.215
13 - 130.85.70.149

Gnutella is a common per-to-peer file sharing service and one you “do not” want on your network. Gnutella can bypass firewalls that are setup to block the file downloads, and that’s not a good thing. As stated on http://www.nwconnection.com/2001_09/gnutel91/, “if the servent that has the desired file does not permit an incoming file download connection on the Gnutella default port, that servent can send a Push request that lists an allowed IP address and a port. Users inside a company can also get around a firewall by setting up the Gnutella application to use ports that are allowed through the firewall, such as port 80 (HTTP), 443 (HTTPS), 23 (Telnet), 25 (SMTP), or 110 (POP3).” It’s recommended that you set up port filtering on ports 6346 and 6347.

Source: http://www.ciac.org/ciac/bulletins/h-109.shtml
http://www.nwconnection.com/2001_09/gnutel91/

Alert 10: suspicious host traffic

Top 5 Source IPs
2,283 - 130.85.157.248 (host)
25 - 204.117.70.7
25 - 204.117.70.5
25 - 130.85.157.252
18 - 195.54.102.4

Top 5 Destination IPs
628 - 130.85.157.248 (host)
38 - 130.85.157.252
20 - 204.117.70.5
17 - 204.117.70.7
10 - 195.54.102.4

Suspicious host traffic is a customized Snort alert rule set used to identify transactions over the network which the system administrator has deemed as malicious in nature or simply worth monitoring.

LINK ANALYSIS

It was interesting to see how Nimda wormed its way though the network. As we know, Nimda can spread itself across the network by several different methods (linked by file sharing, the web, email, etc.) as noted in my paper "Let's Take a Look at NIMDA." Still it was interesting to note the speed with which it infected machine-after-machine. An analysis of the 5 day alert logs identified 65 MY.NET machines triggering (as a source) the "spp_http_decode IIS Unicode attack detected" alerts. While it isn't proof positive that a machine is infected just because it sets this alert off (false positive) any machine triggering this alert as a source (especially a large number of time) should be assumed infected, taken off line immediately and inspected as soon as possible. You will note from the illustration below that on day 1 of the 5 day alerts review 58% of all the machines identified as infected during this analysis were either already infected or became infected on that day.

Sixth five (65) MY.NET source IPs taken from spp_http_decode IIS Unicode attack detected logs:

JUNE 12, 2002
Day 1 – 58%
MY.NET.10.89 →
MY.NET.150.103 →

MY.NET.150.97	→										
MY.NET.151.108	→										
MY.NET.151.64	→										
MY.NET.151.73	→										
MY.NET.152.170	→										
MY.NET.152.180	→										
MY.NET.153.105	→	JUNE 13								JUNE 17	
MY.NET.153.106	→	Day 2 – 71%		JUNE 14			JUNE 15			Day 5, 100%	
MY.NET.153.114	→	MY.NET.150.165	→	Day 3 – 80%			Day 4 – 88%	→	MY.NET.151.18		
MY.NET.153.115	→	MY.NET.152.171	→	MY.NET.152.49	→	MY.NET.152.159	→	MY.NET.152.176			
MY.NET.153.117	→	MY.NET.152.213	→	MY.NET.153.142	→	MY.NET.153.136	→	MY.NET.152.179			
MY.NET.153.118	→	MY.NET.152.22	→	MY.NET.153.173	→	MY.NET.153.145	→	MY.NET.152.186			
MY.NET.153.119	→	MY.NET.152.44	→	MY.NET.153.197	→	MY.NET.153.175	→	MY.NET.152.52			
MY.NET.153.120	→	MY.NET.153.189	→	MY.NET.88.181	→	MY.NET.88.151	→	MY.NET.153.195			
MY.NET.153.123	→	MY.NET.88.140	→	MY.NET.88.201	→		→	MY.NET.88.143			
MY.NET.153.124	→	MY.NET.88.146	→				→	MY.NET.88.149			
MY.NET.153.127	→										
MY.NET.153.148	→										
MY.NET.153.152	→										
MY.NET.153.153	→										
MY.NET.153.157	→										
MY.NET.153.160	→										
MY.NET.153.163	→										
MY.NET.153.166	→										
MY.NET.153.167	→										
MY.NET.153.168	→										
MY.NET.153.176	→										
MY.NET.153.182	→										
MY.NET.153.188	→										
MY.NET.153.193	→										
MY.NET.153.196	→										
MY.NET.153.202	→										
MY.NET.153.203	→										
MY.NET.153.45	→										
MY.NET.153.71	→										
MY.NET.83.90	→										

Prompt identification and corrective action on the part of the system administrator could have reduced, eliminated, or possibly even avoided this virus/worm infection altogether. This poor fellow has a lot of hard work ahead of him/her.

SCAN ANALYSIS

For scan analysis to be effective you need to look at more than just the individual scan. Effective scan analysis requires a knowledge of frequent scans, source and destination hosts, and destination ports. Armed with this knowledge you can keep a watchful eye on your network and react when there is a change in "normal operations." Lists such as the ones below, if maintained over time can help establish what one might refer to as "normal operating procedures/patterns."

Scans by Occurrence

273,370	-	UDP (type of scan)
169,536	-	SYN
96	-	INVALIDACK
75	-	NOACK
72	-	NULL

41	-	UNKNOWN
14	-	VECNA
11	-	FIN
4	-	NMAPID
4	-	FULLXMAS
3	-	XMAS
3	-	SPAU
1	-	SYNFIN

The following is a list of the top 10 source and destination hosts, and destination ports taken from the scans logs. Selection criteria is based on the number of scans associated with the host and port.

Top 10 Sources

110,408	-	MY.NET.150.225 (host)
44,149	-	MY.NET.160.114
37,543	-	213.93.23.218
34,554	-	MY.NET.150.133
23,749	-	MY.NET.98.139
16,226	-	211.184.223.2
16,158	-	205.188.233.153
14,451	-	195.190.34.55
13,940	-	217.58.147.39
11,750	-	MY.NET.150.220

Top 10 Destinations

23,750	-	158.75.57.4 (host)
3,918	-	24.13.123.8
3,289	-	MY.NET.145.197
2,773	-	65.0.39.132
2,594	-	MY.NET.70.92
2,539	-	24.252.125.150
2,358	-	MY.NET.178.154
2,179	-	MY.NET.107.4
2,162	-	MY.NET.108.15
2,154	-	166.84.159.101

Top 10 Destination Ports

134,526	-	28800 (port)
57,153	-	21
30,330	-	6970
27,594	-	27005
20,368	-	23
18,968	-	53
10,679	-	6112
8,622	-	110
8,387	-	24452
8,361	-	7778

The SYN is a very common scan technique and is used as a reconnaissance tool. By simply setting the TCP SYN flag an entire range of IP addresses can be scanned in an instant. Scanning tools like nmap will send invalid flag set combinations to destination hosts and use the hosts response to gain information about it. A common example would be the destination host's operating system. There are many other types of scans which utilize the TCP flag set and several of these listed below are addressed in the SANS Institute Track 3 – Intrusion Detection In-Depth course. Note that these are the same scans as listed above.

INVALIDACK	--	ACK set, not normal, no SPAU or FULLXMAS
NOACK	--	A flag is missing
NULL	--	None of SFRPAU
UNKNOWN	--	Ref. spp_portscan.c source code
VECNA	--	One of the following: P, U, PU, FP, FU
FIN	--	F flag
NMAPID	--	SFPU flags
FULLXMAS	--	SFRPAU flags
XMAS	--	FPU flags
SPAU	--	SPAU flags
SYNFIN	--	SF flags

The top 10 destination ports listed above are defined as follows:

Port 28800: Network gaming, MSN Gaming Zone. Blocking UDP to port 28800 is recommended by firewall manufacturers. With that said it appears to me that we have found the universities most popular online gaming system.

Port 21: File transfer protocol (ftp). The most common exploit of an FTP service is simply the using of FTP to cache files, such as "warez". Never configure an FTP services such that someone can read and write to the same directory. (http://www.iss.net/security_center/advice/Services/FTP/default.htm).

Port 6970: (UDP) RealTime Protocol. (UDP) RealAudio and QuickTime 4 use ports starting at 6970 to send incoming audio streams. College students are going to be using these services and in turn running data through port 6970.

Port 27005: Used in on-line gaming. The game Half-Life is an example of an on-line game that uses this port to communicate with the distant server. It is suspected that due to the large number of university students playing this game online, the intrusion detection software is identifying the data being sent to and from this port as a scan. This port could be blocked if so desired.

Port 23: Telnet is used to remotely log into UNIX machines. It should be noted that telnet has numerous security problems. Keep an eye on this port.

Port 53: Domain name server or DNS (UDP/TCP). Due to its many BIND vulnerabilities this port is a candidate for regular scanning.

Port 6112: "dtspcd" runs on this port. Cert.org, Vulnerability Note VU#172583 at http://www.kb.cert.org/vuls/id/172583 states the following. "A remotely exploitable buffer overflow exists in the Common Desktop Environment (CDE) Subprocess Control Service (dtspcd). An attacker who successfully exploits this vulnerability can execute arbitrary code as root." Recommend blocking or as a minimum restricting external access. Note: some Internet gaming also use this port.

Port 110: Also know as POP 3. Programs like Outlook, Netscape, Eudora, etc. use POP3 to send and receive e-mail from a mail server. POP3 is one of the most popular e-mail services on the Internet, but it is also one hackers love to try to break into.

Port 24452: What we have here is a search for a back door (root shell on port 24452) on a Redhat Linux unpatched box. If the university uses Linux host, it would be wise to use nmap to check for open port number 24452. Additionally, it would also be prudent to map all open ports on all university machines using nmap and follow up with periodic scans looking for changes.

Port 7778: Unreal Tournament, an online multiplayer personal shooter. When gaming, UDP Port 7778, is used to send "heartbeat" packets to the master game server every few minutes. This action lets the master server know the game server is still online. Once again it is suspected that due to the large number of university students playing online games, the intrusion detection software is identifying the data being sent to and from this port as a scan.

OUT-of-SPEC ANALYSIS

The following are all Out-of-Spec (OOS) packets received from external sources.

Source		
7	-	64.4.124.151 (host)
6	-	24.112.58.210
4	-	195.101.94.208
2	-	193.6.40.86
1	-	68.80.114.202
1	-	68.50.107.141
1	-	62.99.143.179
1	-	62.99.143.178
1	-	62.78.169.87
1	-	12.224.236.145
1	-	12.217.65.38

Destination
9 - MY.NET.150.209 (host)
7 - MY.NET.88.165
3 - MY.NET.5.96
3 - MY.NET.150.83
2 - MY.NET.88.162
2 - MY.NET.5.95

Destination Port
8 - 80 (port)
8 - 6346
5 - 1269
2 - 3193
1 - 4106
1 - 2656
1 - 1214

The OOS logs contain a variety of packets which have been crafted. Nine of the 26 OOS packets had the TCP flag 21S***** set. The TCP flag 21S***** set communicates the SYN and reserve bits. These are good packets and a valid use of TCP flag sets.

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
06/12-00:39:43.957516 193.6.40.86:55089 -> MY.NET.150.209:6346
TCP TTL:48 TOS:0x0 ID:13258  DF
21S***** Seq: 0xABA41371   Ack: 0x0   Win: 0x16D0
TCP Options => MSS: 1460 SackOK TS: 2630116 0 EOL EOL EOL EOL

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
06/12-02:43:06.553392 62.99.143.178:59781 -> MY.NET.150.83:80
TCP TTL:47 TOS:0x0 ID:28849  DF
21S***** Seq: 0x82D3E70E   Ack: 0x0   Win: 0x16D0
TCP Options => MSS: 1460 SackOK TS: 377001919 0 EOL EOL EOL EOL

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

Removing the good TCP flag 21S***** packets will leave the following OOS packets for analysis.

Source
7 - 64.4.124.151 (host)
6 - 24.112.58.210
1 - 68.80.114.202
1 - 68.50.107.141
1 - 12.224.236.145
1 - 12.217.65.38

Destination

7	-	MY.NET.88.165 (host)
6	-	MY.NET.150.209
2	-	MY.NET.5.96
2	-	MY.NET.88.162

Destination Port

5	-	6346 (port)
5	-	1269
2	-	3193
2	-	80
1	-	4106
1	-	2656
1	-	1214

A review of these packets shows the top source IP 64.4.124.151 always going to destination IP MY.NET.88.165 using ports 1269 and 3193. Additionally, these packets all occurred on June 13, between 17:39 and 18:36, a time span of slightly less than a minute.

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
06/13-17:39:58.851407 64.4.124.151:3193 -> MY.NET.88.165:1269
TCP TTL:113 TOS:0x0 ID:52404  DF
21**R**U Seq: 0xBCCA1D8   Ack: 0x7D86   Win: 0x5010
0C 79 04 F5 0B CC A1 D8 00 00 7D 86 04 E4 50 10  .y........}...P.
79 34 A9 3E 00 00 C2 31 19 C0 20 0F B0 1A 62 7A  y4.>...1.. ...bz
F3 93                                            ..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
06/13-17:46:22.699466 64.4.124.151:0 -> MY.NET.88.165:3193
TCP TTL:113 TOS:0x0 ID:61990  DF
21**RP*U Seq: 0x4F50D80   Ack: 0x1D87D87   Win: 0x5010
3C EC 50 10 7B 30 F7 71 00 00 3E FA 61 41 AF A4  <.P.{0.q..>.aA..
76 86 A2 1B F5 D2                                v.....

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
06/13-17:54:54.956901 64.4.124.151:4 -> MY.NET.88.165:3193
TCP TTL:113 TOS:0x0 ID:65215  DF
21**R*** Seq: 0x4F50FC7   Ack: 0x31D87D88   Win: 0x5010
TCP Options => EOL EOL EOL EOL EOL EOL SackOK SackOK SackOK EOL
Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53
Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53
Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53 Opt 53

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
06/13-18:00:01.789438 64.4.124.151:3193 -> MY.NET.88.165:1269
TCP TTL:113 TOS:0x0 ID:21275  DF
21**R**U Seq: 0x11122   Ack: 0xD1D87D89   Win: 0x5010
```

```
TCP Options => EOL EOL

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
06/13-18:16:02.185414 64.4.124.151:3193 -> MY.NET.88.165:1269
TCP TTL:113 TOS:0x0 ID:1338  DF
21**RP*U Seq: 0x1566B78C   Ack: 0x7D8C   Win: 0x5010
TCP Options => EOL EOL Opt 23 (3): 1FFC Opt 252

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
06/13-18:18:59.662527 64.4.124.151:3193 -> MY.NET.88.165:1269
TCP TTL:113 TOS:0x0 ID:15215  DF
*1SF**** Seq: 0x163141D8   Ack: 0x7D8D7EAC   Win: 0x5010
0C 79 04 F5 16 31 41 D8 7D 8D 7E AC 00 83 50 10  .y...1A.}.~...P.
78 30 9D E1 00 00 C7 DE 40 04 C4 CE 52 1C DE 7D  x0......@...R..}
D0 01                                            ..

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=
06/13-18:36:41.669086 64.4.124.151:3193 -> MY.NET.88.165:1269
TCP TTL:113 TOS:0x0 ID:37288  DF
21**RP*U Seq: 0x1ADFD1D8   Ack: 0x927D90   Win: 0x5010
TCP Options => EOL EOL

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

An examination of the logs showed that source IP 64.4.124.151 triggered 23 alerts in the alerts logs. Out-of Spec packet analysis revealed multiple TCP flag set irregularities. Further analysis identified this IP as an inactive host which means it is probably turned off and is a catch-as-catch-can IP. It would be wise to put this IP on a watch list and see if it continues. A Whois query revealed the following results.

```
Search results for: 64.4.124.151
OrgName:    Ntelos Inc.
OrgID:      NTELO

NetRange:   64.4.96.0 - 64.4.127.255
CIDR:       64.4.96.0/19
NetName:    NTELO-BLK-2
NetHandle:  NET-64-4-96-0-1
Parent:     NET-64-0-0-0-0
NetType:    Direct Allocation
NameServer: ns1.ntelos.net
NameServer: ns2.ntelos.net
Comment:    ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE
RegDate:    2001-03-20
Updated:    2002-10-03

OrgName: Ntelos Inc.
```

OrgID: NTELO
Address: 1154 Shenandoah Village Dr
Floor #3 Waynesboro VA 22980
Country: US
Comment:
RegDate: 2002-08-25
Updated: 2002-08-26

Analysis of the destination IP MY.NET.88.165 showed that it did not set off any alerts in the alerts log.

ANALYSIS PROCESS

I followed Joe Ellis' (http://www.giac.org/practical/Joe_Ellis_GCIA.doc) suggestion of using Unix-based text processing tools wc, uniq, grep, sed, awk, cat, cut, and sort. This approach proved most helpful in manipulating data the files required for this analysis. As did he, I too recommend using these Unix tools to assist in log analysis.

Data manipulation for the alerts, scans, and out-of-spec analysis was done at my place of work using a Unix machine. The research and writing of this paper was done on my home computer using a Windows PC.

The following commands are based on Joe Ellis's GCIA paper. However, to make his commands work for me I had to modify them to meet my needs.

Unix commands used for Alerts analysis.

Move all alert files into a single file for processing:
Cat alert. > alert.txt

*Cleanup Alerts - Remove portscans and "start" and "stop" comments leaving only alerts:
Cat alert.txt | sed –e 's/\[**\]/**/g' | egrep –v "(spp_port_scan|Null scan|SCAN Synscan Portscan)" > sed.out

*Sort alerts by occurrence:
Cat sed.out | awk –F'**' '{print $3}' | sort | uniq –c | sort –nr > top10.txt

*List of source addresses sorted by occurrence:
Cat sed.out | awk –F'**' '{print $3}' | awk –F '->' '{print $1}' awk –F':' '{print $1}' | sort | uniq –c | sort –nr > top10src_noscans.txt

*List of destination addresses sorted by occurrence:
Cat sed.out | awk –F'**' '{print $3}' | awk –F '->' '{print $2}' awk –F':' '{print $1}' | sort | uniq –c | sort –nr > top10dst_noscans.txt

*Commands used to sort source and destination addresses by alert and occurrence:
Grep ‘alert name’ sed.out | awk –F’**’ ‘{print $3}’ | awk –F’->’ ‘{print $1}’ | awk –F’:’ ‘{print $1}’ | sort | uniq –c | sort –nr > alert_name_src.txt

Grep ‘alert name’ sed.out | awk –F’**’ ‘{print $3}’ | awk –F’->’ ‘{print $2}’ | awk –F’:’ ‘{print $1}’ | sort | uniq –c | sort –nr > alert_name_dst.txt

Unix commands used for Scans analysis:

*Move all scan files into a single file for processing:
Cat scan* > scans.txt

*Cleanup scans:
Grep ‘Jun’ scans.txt | awk ‘{print $4,”\t”,$6,”\t”,$7,”\t”,$8}’ > allscans.txt

*List of scans sorted by occurrence:
Cat allscans | awk ‘{print $3}’ | sort | uniq –c | sort –nr > top10_scans.txt

*List of scan source addresses sorted by occurrence:
awk ‘{print $1}’ allscans | awk –F’:’ ‘{print $1}’ | sort | uniq –c | sort –nr > src_scans.txt

*List of scan destination addresses sorted by occurrence:
awk ‘{print $2}’ allscans | awk –F’:’ ‘{print $1}’ | sort | uniq –c | sort –nr > dst_scans.txt

*List of scan destination ports sorted by occurrence:
awk ‘{print $2}’ allscans | awk –F’:’ ‘{print $2}’ | sort | uniq –c | sort –nr > dst_port.txt

Unix commands used for Out-of-Spec (OOS) analysis.

*List of OOS destination addresses sorted by occurrence:
Grep “..\/..\-..\:..\:” oos.txt | cut –d \> -f2 | cut –f1 –d” ” | sed ‘s/\ //g’ | sort | uniq –c | sort –nr > oos_dst_ips.txt

*List of OOS destination ports sorted by occurrence:
Grep “..\/..\-..\:..\:” oos.txt | cut –d \> -f2 | cut –f2 –d” ” | sed ‘s/\ //g’ | sort | uniq –c | sort –nr > oos_dst_ports.txt

*List of source addresses sorted by occurrence:
Grep “..\/..\-..\:..\:” oos.txt | cut –d \> -f1 | cut –f2 –d” “ | sed ‘s/\ //g’ | sort | uniq –c | sort –nr > oos_src_ips.txt

REFERENCES

1. Jim Forster - neohapsis.com, Re: [snort] 'SMB Name Wildcard'
URL: http://archives.neohapsis.com/archives/snort/2000-01/0222.html
2. Bryce Alexander - SANS Institute, Port 137 Scan
URL: http://www.sans.org/newlook/resources/IDFAQ/port_137.htm
3. Carnegie Mellon Software Engineering Institute, SNMP Grabbing
URL: http://ki.sei.cmu.edu/idar/drill_attack.cfm?attack=SNMP%20Grabbing
4. Carnegie Mellon CERT Coordination Center, Vulnerabilities in Simple Network Management Protocol (SNMP)
URL: http://www.cert.org/advisories/CA-2002-03.html
5. Unicode Inc., “What is Unicode?”
URL: http://www.unicode.org/unicode/standard/WhatIsUnicode.html
6. F-Secure Virus Descriptions, “Nimda”
URL: http://www.f-secure.com/v-descs/nimda.shtml
7. Ellen Messmer and John Fontana – Network World Fusion, Update: “Nimda worm spreads three ways; seen as major threat”
URL: http://www.nwfusion.com/news/2001/0918admindll.html
8. Ellen Messmer and John Fontana - Network World Fusion, “Nimda virus riles up Microsoft users”
URL: http://www.nwfusion.com/news/2001/0924nimdams.html
9. D. E. Levine - WinPlanet, “Handling Nimda”
URL: http://www.winplanet.com/winplanet/tips/3787/1
URL: http://www.winplanet.com/winplanet/tips/3787/2
10. Dean Chetkovich - California State Polytechnic University, “NIMDA Virus Warning”
URL: http://www.csupomona.edu/~housing/update102.htm
11. Robert Vamosi - California State Polytechnic University, “How it works (Nimda)”
URL: http://www.csupomona.edu/~housing/update102.htm
12. Robert Lemos - California State Polytechnic University, “Lethal worm spells double trouble”
URL: http://www.csupomona.edu/~housing/update102.htm
13. Carnegie Mellon CERT Coordination Center, “CERT® Advisory CA-2001-26 Nimda Worm”
URL: http://www.cert.org/advisories/CA-2001-26.html
14. Norman - The Antivirus Company, “W32/Nimda.A@mm”
URL: http://www.norman.no/virus_info/w32_nimda_a_mm.shtml
15. GFI Software Ltd, “NEWS/Nimda worm: Description”
URL: http://www.gfi.com/news/press.asp?release=nimdaworm&lcode=en
16. Tjerk Vonck & mIRC Co. Ltd., “What is IRC?”
URL: http://www.mirc.com/irc.html
17. Newircusers.com, “Internet Relay Chat/ Instant Messenger Chats”
URL: http://www.newircusers.com/java/index.html
18. Newircusers.com, “IRC and Internet Security”
URL: http://www.newircusers.com/java/security.html
19. Whitehats, Inc., “IDS311 "PING-SCANNER-L3RETRIEVER"”

URL: http://www.whitehats.com/info/IDS311
20. Steven L. Drew 0530 - GCIA Intrusion Detection In Depth, May 14, 2002
URL: http://www.giac.org/practical/Steven_Drew_GCIA.doc
21. Kevin Timm 0526 - GCIA Intrusion Detection In Depth, March 10 – 16 2002
URL: http://www.giac.org/practical/Kevin_Timm_GCIA.doc
22. Joe Ellis 0522 - GCIA Intrusion Detection In Depth, May 14, 2002
URL: http://www.giac.org/practical/Joe_Ellis_GCIA.doc
23. Pedro Bueno 0518 - GCIA Intrusion Detection In Depth
URL: http://www.giac.org/practical/Pedro_Bueno_GCIA.doc
24. Tod A. Beardsley 0525 - GCIA Intrusion Detection In Depth, May 8, 2002
URL: http://www.giac.org/practical/Tod_Beardsley_GCIA.doc
25. Nmap.com, “What is nmap?”
URL: http://www.nmap.org/nmap/index.html#intro
26. Microsoft Corporation - MSN Messenger
URL: http://messenger.msn.com/
27. Microsoft Corporation - MSN Messenger
URL: http://messenger.msn.com/support/features.asp?client=1
28. Symantec Corp, “W32.Choke.Worm”
URL: http://www.symantec.com/avcenter/venc/data/w32.choke.worm.html
29. CIAC - U.S. Department of Energy, “H-10-9: Solaris DCE and AFS Integrated login Vulnerability”
URL: http://www.ciac.org/ciac/bulletins/h-109.shtml
30. Laura Chappell – Novell Connection, “Just Say Gno! (Gnutella)”
URL: http://www.nwconnection.com/2001_09/gnutel91/
31. Internet Security Systems, “FTP”
URL: http://www.iss.net/security_center/advice/Services/FTP/default.htm
32. Carnegie Mellon CERT Coordination Center, “CDE Subprocess Control Service dtspcd contains buffer overflow”
URL: http://www.kb.cert.org/vuls/id/172583
33. TurSecure Corp – ICSA Labs, “An Introduction to Intrusion detection Assessment”
URL: http://www.icsalabs.com/html/communities/ids/whitepaper/Intrusion1.pdf
34. Hosted by Sourcefire, Inc. - Snort IDS
URL: http://www.snort.org
35. SANS Institute, Information security Reading Room, “Using Snort v.18 with SnortSnarf on a RedHat Linux System”
URL: http://rr.sans.org/intrusion/snortsnarf.php
36. Carnegie Mellon CERT Coordination Center, “Analysis Console for Intrusion Database (ACID)”
URL: http://www.cert.org/kb/acid/
37. Berkers, John - Open Source Development Network, Inc., “IIS Unicode attack detected”.
URL: http://www.geocrawler.com/mail/msg.php3?msg_id=6390557&list=4890
38. IBM Corp – IBM Pittsburgh Lab, “UDP Ports Used by AFS”
URL: http://www.transarc.ibm.com/Support/afs/admin/UDP.html