Global Information Assurance Certification Paper

Intrusion Detection in Depth
GCIA Practical Assignment
Version 3.0

David S. Dobrotka


Intrusion Detection in Wireless Networks

Introduction

The IEEE 802.11b wireless LAN standard has become the de-facto for wireless network communications medium. The availability of inexpensive equipment coupled with wire-like network bandwidth and ease of use has driven rapid adoption by corporate and SOHO users. Unfortunately, the rapid implementation of this wireless technology is having other, unintended consequences.

Historical Context

Brought into the public eye 20 years ago by the movie War Games, war dialing, or systematically dialing ranges of phone numbers to discover computer systems, continues to plague corporate America. War dialing will often discover modems attached to corporate desktops, which are in turn connected to the corporate LAN. These computers are often loaded with remote control software, such as PCAnywhere or Carbon Copy, allowing the individual dialing the modem to control the remote computer as if they were sitting at the keyboard.

A similar situation is beginning to develop with wireless technology. Business units or individuals install wireless access points (AP), acting as bridges to the corporate LAN, broadcasting to anyone with a $50 wireless network card and a laptop, up to three football fields distant. "War driving," like its cousin described above, allows those with the tools described here to find, catalog, and access vulnerable wireless APs, and possibly gain access to any physically connected network, from the relative anonymity of a rental car in the parking lot.

Objective

The scenario described above is but one of the threats which an intrusion detection analyst must consider. First, however, we must ask a more fundamental question: what is intrusion detection when applied to wireless networks? Intrusion detection systems collect information about observable or auditable events, which are then analyzed and correlated to determine things like cause or motive. Therefore, in order to provide a basis for wireless intrusion detection, we must first determine what can be observed and collected for analysis. This paper will discuss several rudimentary events which could be captured by a wireless intrusion detection system and present a survey of tools to accomplish those tasks.

This is not your father's network

Current intrusion detection solutions rely on the relatively static and contained nature of wired networks. Potential intruders would need to gain physical access to a network jack or logically enter the network through well-defined pathways. Placing intrusion detection sensors was a matter of finding (or creating) places where all or most network traffic transited. These assumptions are no longer valid for wireless networks.

The IEEE 802.11 standard [1] defines two types of wireless network topologies: Independent Basic Service Set (IBSS, or "ad hoc"), and Basic Service Set (BSS, or "infrastructure"). The IBSS topology involves two or more wireless stations communicating peer-to-peer (Figure 1).

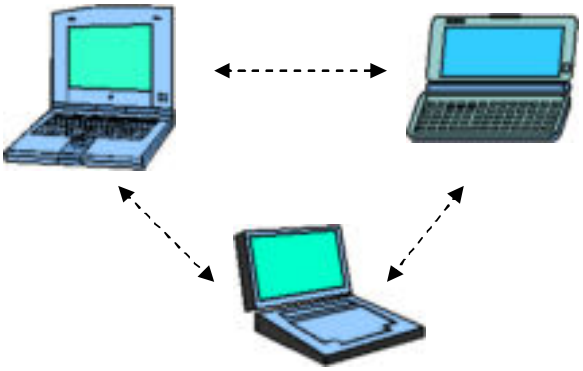


Figure 1: Ad-hoc

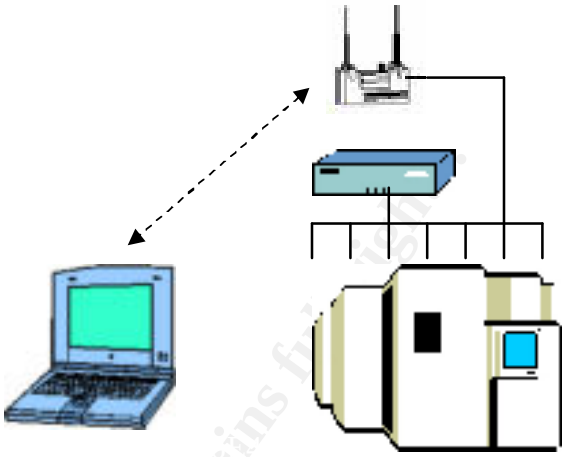


Figure 2: Infrastructure

The BSS topology (Figure 2) adds an AP attached to a "distribution system" (usually a network, like Ethernet); all communications route through the AP.

An ad hoc network has some obvious disadvantages for intrusion detection. Yongguang Zhang and Wenke Lee have written an excellent paper [2] addressing this particular problem. They outline several fundamental issues with wireless ad hoc networks:

- Wireless stations are all independent nodes. Each must be responsible for it's own protection from attack and compromise. Compromising only one node or introducing a malicious node may affect the viability of the entire network.
- No central point exists from which to monitor all network traffic.
- Differences between normal and anomalous traffic patterns may be indistinguishable. The mobile nature of the wireless stations can make legitimate network traffic appear suspect.

Zhang and Lee propose an architecture in which all nodes act as independent IDS sensor; able to act independently and cooperatively. Events are generated from a local detection engine. If analysis of the events are inconclusive or require more information, other, local sensors can be utilized. Each independent sensor has six modules, three of which pertain to intrusion detection:

- data collection: the types of raw data the detection engines will utilize includes system and user activities, local communication activities, and "observable" (nodes within range) communications activities
- local detection – since it will be difficult to maintain and distribute an anomalous signature database, Zhang and Lee propose defining statistically "normal" activities specific to each node, which will therefore reside locally on each node.
- cooperative detection. If the local detection engine does not have enough evidence to alert on a suspected problem, it can ask other nodes for assistance. Information describing the detect gets propagated to neighboring nodes. Evidence returned from neighboring nodes can be used to create a new evaluation of the detect.

Infrastructure mode is where current intrusion detection methodologies and collection techniques become useful. Since all traffic transits through the AP, close proximity to the AP becomes a logical choice to place a sensor. Since 802.11b is essentially just another physical medium, the

AP acts as a bridge – translating 802.11b frames to 802.3 (or some other network medium) frames, and vice versa. Data encapsulated at higher layers is unchanged. To collect events of interest at Layer 3 and above, one can rely on current tools, such as tcpdump. To look at frame information, however, each tool must be able to interpret the medium frame type.

Events of Interest

Several events of interest would be of obvious interest to an analyst monitoring an access point (it is assumed the reader has sufficient knowledge of the 802.11 standard; please see [1] and [3] for details).

General MAC Frames

Like IP packets, 802.11 frames [1] carry enough useful information to warrant monitoring.

(Octets) 2	**2**	**6**	**6**	**6**	**2**	**6**	**0-2312**	**4**
Frame Control	Duration ID	Address 1	Address 2	Address 3	Sequence Control	Address 4	Frame Body	FCS

MAC Header

2	**2**	**4**	**1**	**1**	**1**	**1**	**1**	**1**	**1**	**1**
Protocol Ver	Type	Subtype	To DS	From DS	More Frag	Retry	Pwr Mgt	More Data	WEP	Order

Beacon Frames (Type 00, Subtype 1000)

"Beacon" frames are regularly transmitted by the AP, and contain information needed by a wireless station to begin the association/authentication process. An analyst may wish to analyze these frames to monitor for rogue access points or other potentially malicious traffic.

Capturing beacon frames is similar to sniffing network traffic on an Ethernet segment. The network card must be in promiscuous mode, but does not necessarily need to have a network address assigned to it. It can capture data, but is virtually invisible to everyone else on the network. The following tools are specifically built for this task (your mileage may vary):

airosniff: http://gravitino.net/~bind/code/airosniff/	Airosniff can be used to assist in the identification of wireless networks by sniffing SSIDs. Airosniff, for the Cisco Aironet card allows one to seek out wireless networks, auto-config the card for sniffing and perform access point vendor identification.
netstumbler:	Windows-based AP discovery tool; excellent

http://www.netstumbler.com/	GUI, easy to use
Wavelan-tools http://sourceforge.net/projects/wavelan-tools/	802.11 network tools - allow for detection of networks and services initially using wireless extensions for linux (openbsd porting simple?) and raw 802.11 frames. initial support is for the wavelan/orinoco card and plan support for aironet cards
bsd-airtools http://www.dachb0den.com/projects/bsd-airtools.html	bsd-airtools is a package that provides a complete toolset for wireless 802.11b auditing. Namely, it currently contains a bsd port of airsnort for wep cracking (as well as kernel patches for NetBSD, OpenBSD, and FreeBSD). It also contains a curses based ap detection application similar to netstumbler (dstumbler) that can be used to detect wireless access points, view signal to noise graphs, and interactively scroll through your scanned ap's and view statistics for each. And recently, we've added a couple new tools to provide a complete toolset for making use of all 14 of the prism2 debug modes as well as do basic analysis of the hardware-based link-layer protocols provided by prism2's monitor debug mode.
APTools http://aptools.sourceforge.net	APTools is a utility that queries ARP Tables and Content-Addressable Memory (CAM) for MAC Address ranges associated with 802.11b Access Points. It will also utilize Cisco Discovery Protocol (CDP) if available. If a Cisco Aironet MAC Address is identified, the security configuration of the Access Point is audited via HTML parsing.
IBM Wireless Security Auditor http://www.research.ibm.com/gsal/wsa/	WSA is an IBM research prototype of an 802.11 wireless LAN security auditor, running on Linux on an iPAQ PDA. WSA automatically audits a wireless network for proper security configuration, to help network administrators close any vulnerabilities before the hackers try to break in. WSA is not a packet dump/analyzer; it does all the necessary packet monitoring and analysis, and provides the user with just the answers to the important management questions. The results are color coded (green is good, red is bad) for rapid and easy understanding.

(Thanks to http://www.personaltelco.net/index.cgi/WirelessSniffers)

Association and Authentication (Type 00)

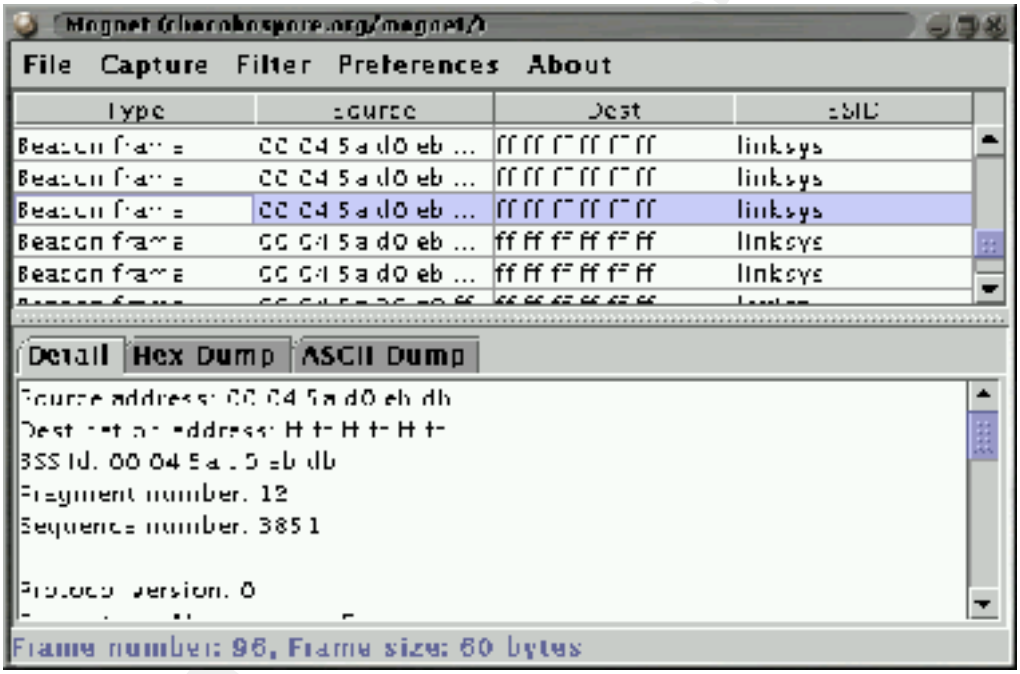
Once an attacker collects an SSID, they may wish to return and actually use the wired network your AP is attached to. In order to do that, the attacker must begin the association and authentication process.

The first frame to be sent by the wireless station is an Association Request Management frame (subtype 0000), to which the AP responds with an Association Response Management frame (subtype 0001). The association response frame contains a 2-byte status code – "0" means success, while all others indicate a problem. Also, the attacker's MAC address has been transmitted over the wireless medium. Analyzing association/authentication response codes and capturing MAC addresses would also be a good basis for intrusion detection events. 802.11b packet analysis tools are now required to capture and display this information. Tools which perform this function include:

Ethereal (Linux or FreeBSD) http://www.ethereal.com/	Ethereal is a GUI sniffer which understands 802.11b frames.
Mognet http://chocobospore.org/mognet/	Mognet is a free wireless ethernet sniffer/analyzer written in Java Features include: • Realtime output of captured frames. • Detailed description of any frame, including all IEEE 802.11 generic and frame-specific headers. • Raw hex and ascii dump of any frame. • Space-efficient presentation of information for convenient operation on handhelds.
Airopeek from Wild Packets (Windows) http://www.wildpackets.com/products/airopeek	"Airopeek is a comprehensive packet analyzer for IEEE 802.11b wireless LANs, supporting all higher level network protocols such as TCP/IP, Appletalk, NetBEUI, and IPX. Affordable and easy-to-use, Airopeek contains all of the network troubleshooting features familiar to users of our award-winning Etherpeek. In addition, Airopeek quickly isolates security problems, fully decodes 802.11b WLAN protocols, and expertly analyzes wireless network performance with accurate identification of signal strength, channel and data rates."
Sniffer Wireless from Cisco (Windows)	"Sniffer Wireless was designed in

http://www.sniffer.com/products/wireless/default.asp?A=5	accordance with the IEEE 802.11b interoperability standard. It includes network monitoring, capturing, decoding, and filtering-all the standard award-winning Sniffer Pro features you already know and appreciate. Sniffer Wireless also provides the most comprehensive 802.11b solution to the unique aspects of wireless networks. Sniffer Wireless is the industry-first Wireless LAN management tool that can spot security risks in real-time, identify network problems efficiently and reduce network-operating costs."

Here is Mognet in action:



ARP

The address resolution protocol (ARP) is used to map an IP address to a corresponding hardware address [4]. Arpwatch (http://www-nrg.ee.lbl.gov/) is a tool which monitors changes to this information and can be used as a source of detection data. When applied to a wireless access point [5], arpwatch could be used to obtain information about wireless stations already authenticated and associated with the AP. Once a packet enters the wired side of the AP from the wireless side, interesting traffic may begin to appear. For example, Richard Johnson (http://www.monkey.org/openbsd/archive/ports/0012/msg00098.html) noted:

You'll see a lot of the following if you're watching ARPs from across an 802.11b wireless bridge to a 10baseT LAN or the like:

```
ethernet mismatch
The source mac ethernet address didn't match the
address inside the arp packet.
```

The source MAC addr on the packet will of course be that of the wireless<->ethernet bridge, while the MAC addr inside the packet will be the other host's actual ethernet MAC addr.

Analysis

Due to the large amount of raw data that will be collected by these tools, the analyst will be forced to develop procedures to reduce it. Statistical methods must be employed to bring order to the data. The anomaly detection routines described by Zhang and Lee [2] and XXX could be applied here. For example, given a fixed time period, tally the number of Association/Authentication requests and Association/Authentication response status codes and corresponding MAC address for typical network situations. This is called a "normal profile" in [2]. Other profiles not fitting this typical profile can be alerted to the analyst. In this case, a large number of Authentication response codes of 15 (Authentication rejected because of challenge failure), over a short period of time (both "large" and "short" will be defined by the normal profile), with the same source MAC address, should generate an event.

Conclusion

The process, methodology, and tools described above simply scratch the surface of wireless intrusion detection. This paper has described the most rudimentary form of wireless intrusion detection for the most basic network architecture– detecting wireless stations associating with an access point attached to a wired network. Much more work needs to be done develop the state of wireless intrusion detection. With the increasing popularity of "war driving", this capability will certainly be required to help protect our wireless infrastructure.

© SANS Institute 2000 - 2002 As part of GIAC practical repository. Author retains full rights.

References

1. ANSI/IEEE. "IEEE Standard for Information technology. Telecommunications and information exchange between systems. Local and metropolitan area networks. Specific requirements. Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications." 1999.
2. Zhang, Yongguang, Wenke, Lee. "Intrusion Detection in Wireless Ad-Hoc Networks." Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking. 2000.
3. Arbaugh, William A., Shankar, Narendar, Wan, Y.C. Justin. "Your 802.11 Wireless Network has No Clothes." Department of Computer Science, University of Maryland. 31 March 2001.
4. Stevens, W. Richard. TCP/IP Illustrated, Volume 1. Massachusetts: Addison-Wesley. 1994.
5. Shipley, Peter. Interview. http://www.starkrealities.com/shipley.html. 31 July 2001.


GCIA Practical Assignment
Network Detects

Detect #1: Suspicious Port Scan

One of the most important skills an intrusion analyst can learn is what "typical" traffic looks like on networks they must monitor. This network normally receives 10-20 scans for various ports on any given day; usually a specific port coupled with selected IP addresses. This scan fell outside the range of "normal," scanning one host for 127 TCP ports.

[**] [100:1:1] spp_portscan: PORTSCAN DETECTED from 212.69.162.53 (THRESHOLD 4 connections exceeded in 0 seconds) [**]
10/29-15:38:11.077704

```
[**] [100:2:1] spp_portscan: portscan status from 212.69.162.53: 8
connections across 1 hosts: TCP(8), UDP(0) [**]
10/29-15:38:14.627012

[**] [100:2:1] spp_portscan: portscan status from 212.69.162.53: 9
connections across 1 hosts: TCP(9), UDP(0) [**]
10/29-15:38:19.646791

 [omitted]

[**] [100:3:1] spp_portscan: End of portscan from 212.69.162.53: TOTAL
time(94s) hosts(1) TCP(127) UDP(0) [**]
10/29-15:39:51.916856
```

Type of Event Generator
SNORT v1.8.1 with slightly modified ruleset, running on NetBSD 1.5.2 i386 (Intel Pentium II), attached to 384Kb line. The SNORT preprocessor generated this alert when more than 4 connection attempts occurred.

Probability the Source Address was Spoofed:
Simply considering the SNORT alert events would lead one to believe spoofing is possible. A malicious user could spoof packets from the same network segment as the suspect host, then sniff the SYN-ACK reply. However, the tcpdump traces below definitely make spoofing a low possibility.

Description of Attack:
The SNORT logs detail what appears to be a directed portscan against a single IP address. If an attacker has previously performed reconnaissance on a target network, they may only subsequently scan IP addresses which are active. To gather a complete picture, SHADOW hourly tcpdump files were pulled to understand the original stimulus of this event.

```
% tcpdump -r tcp.2001102914 host 212.69.162.53

14:38:02.576299 MY.NET.107.66.54994 > onyx.mycgiserver.com.www: S
365032349:365032349(0) win 16384 <mss 1460,nop,nop,sackOK>
14:38:02.748176 onyx.mycgiserver.com.www > MY.NET.107.66.54994: S
880180218:880180218(0) ack 365032350 win 33232 <mss 536>
14:38:02.748766 MY.NET.107.66.54994 > onyx.mycgiserver.com.www: . ack 1 win 0
```

```
14:38:02.749105 MY.NET.107.66.54994 > onyx.mycgiserver.com.www:
. ack 1 win 16616
```

A three way handshake establishes a session over port 80.

```
14:38:02.751048 MY.NET.107.66.54994 > onyx.mycgiserver.com.www: P 1:441(440)
ack 1 win 16616
14:38:02.946807 onyx.mycgiserver.com.www > MY.NET.107.66.54994: . ack 441 win
33232
14:38:03.916524 onyx.mycgiserver.com.www > MY.NET.107.66.54994: P 1:504(503)
ack 441 win 33232
14:38:04.090590 MY.NET.107.66.54994 > onyx.mycgiserver.com.www: . ack 504 win
16113
14:38:05.455306 onyx.mycgiserver.com.www > MY.NET.107.66.54994: P
504:678(174) ack 441 win 33232
14:38:05.593184 MY.NET.107.66.54994 > onyx.mycgiserver.com.www: . ack 678 win
16616
14:38:07.032061 onyx.mycgiserver.com.www > MY.NET.107.66.54994: P
678:851(173) ack 441 win 33232
14:38:07.195018 MY.NET.107.66.54994 > onyx.mycgiserver.com.www: . ack 851 win
16443
14:38:08.588758 onyx.mycgiserver.com.www > MY.NET.107.66.54994: P
851:1024(173)ack 441 win 33232
14:38:08.697175 MY.NET.107.66.54994 > onyx.mycgiserver.com.www: . ack 1024
win 16270
```

Information is passed between HTTP server and web browser.

```
14:38:09.445233 MY.NET.107.66.54994 > onyx.mycgiserver.com.www: R
365032790:365032790(0) win 0
```

The connection is RESET, but the port scan begins.

```
14:38:10.588173 onyx.mycgiserver.com.4212 > MY.NET.107.66.ftp: S
890053244:890053244(0) win 32768 <mss 536,nop,wscale 0>
14:38:10.588669 onyx.mycgiserver.com.4211 > MY.NET.107.66.netbios-ssn: S
889989244:889989244(0) win 32768 <mss 536,nop,wscale 0>
14:38:10.589186 onyx.mycgiserver.com.4213 > MY.NET.107.66.ssh: S
890117244:890117244(0) win 32768 <mss 536,nop,wscale 0>
14:38:10.591261 onyx.mycgiserver.com.4214 > MY.NET.107.66.telnet: S
890181244:890181244(0) win 32768 <mss 536,nop,wscale 0>
14:38:10.595371 onyx.mycgiserver.com.4215 > MY.NET.107.66.smtp: S
890245244:890245244(0) win 32768 <mss 536,nop,wscale 0>
14:38:11.596633 onyx.mycgiserver.com.4217 > MY.NET.107.66.domain: S
891457349:891457349(0) win 32768 <mss 536,nop,wscale 0>
14:38:13.094361 onyx.mycgiserver.com.4220 > MY.NET.107.66.finger: S
893028308:893028308(0) win 32768 <mss 536,nop,wscale 0>
14:38:14.596353 onyx.mycgiserver.com.4224 > MY.NET.107.66.www: S
894834058:894834058(0) win 32768 <mss 536,nop,wscale 0>
14:38:16.099462 onyx.mycgiserver.com.4225 > MY.NET.107.66.81: S
896485219:896485219(0) win 32768 <mss 536,nop,wscale 0>
14:38:16.491209 onyx.mycgiserver.com.4212 > MY.NET.107.66.ftp: S
890053244:890053244(0) win 32768 <mss 536,nop,wscale 0>
14:38:16.491489 onyx.mycgiserver.com.4213 > MY.NET.107.66.ssh: S
890117244:890117244(0) win 32768 <mss 536,nop,wscale 0>
```

```
[omitted]

14:39:18.850259 onyx.mycgiserver.com.4381 > MY.NET.107.66.8010: S
966428384:966428384(0) win 32768 <mss 536,nop,wscale 0>
14:39:20.143117 onyx.mycgiserver.com.4393 > MY.NET.107.66.8080: S
969444574:969444574(0) win 32768 <mss 536,nop,wscale 0>
14:39:20.145511 onyx.mycgiserver.com.4330 > MY.NET.107.66.swat: S
938748629:938748629(0) win 32768 <mss 536,nop,wscale 0>
14:39:44.843396 onyx.mycgiserver.com.4393 > MY.NET.107.66.8080: S
969444574:969444574(0) win 32768 <mss 536,nop,wscale 0>
```

127 ports are scanned in 94 seconds. Attackers will generally do a more complete port scan, such as this, if they are 1) script kiddies, or 2) trolling for more information about a given host. This type of information gathering exercise may foreshadow a more intense campaign against your host.

Attack Mechanism

Since the stimulus for this attack appeared to be a fully-established HTTP session, a hex dump of the packets in question should reveal where the browser was headed.

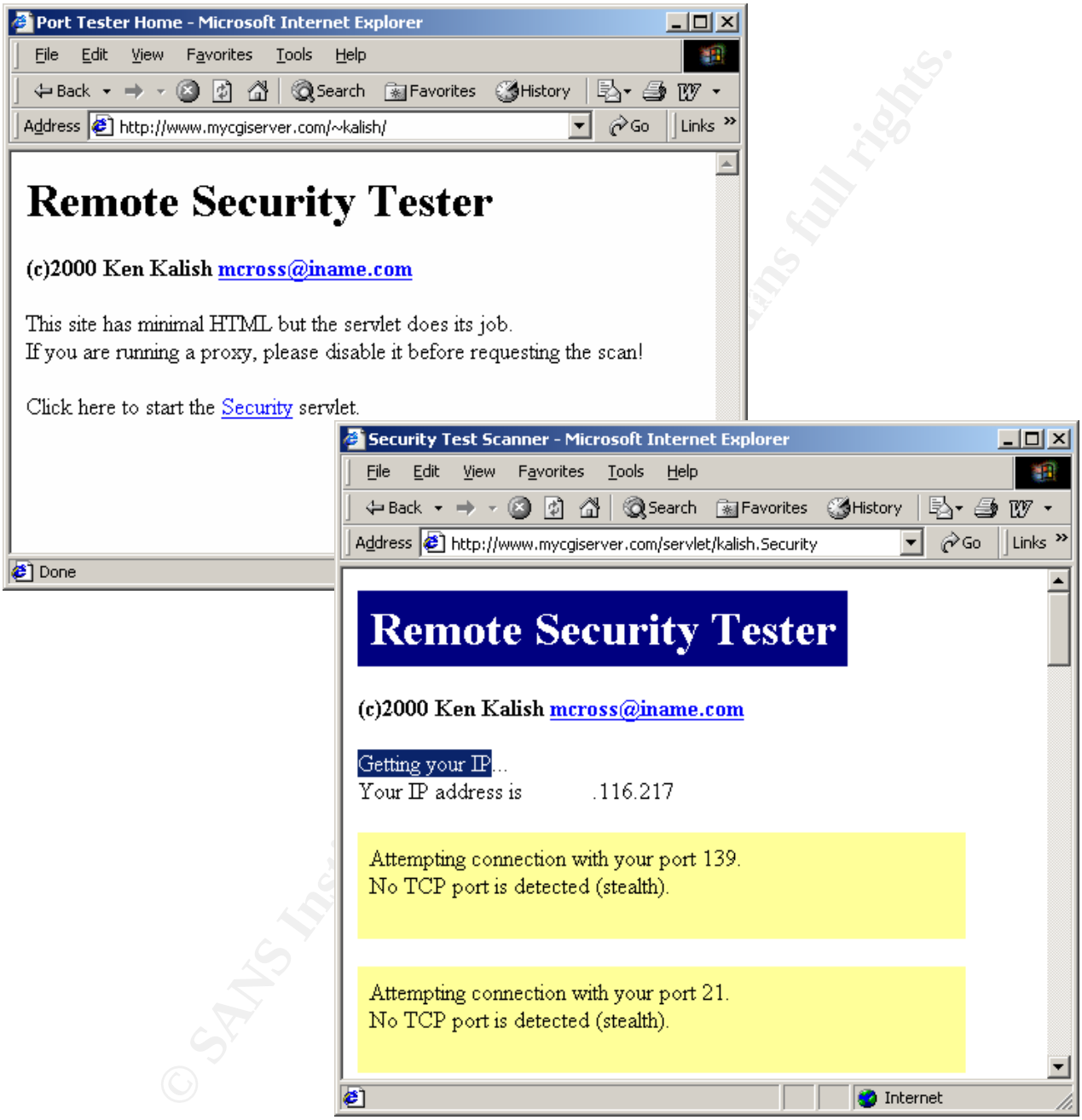
```
%tcpdump -x -X -r tcp.2001102914 host 212.69.162.53

14:38:02.751048 MY.NET.107.66.54994 > onyx.mycgiserver.com.www:
P 1:441(440) ack 1 win 16616
0x0000   4500 01e0 2146 0000 7f06 f738 xxyy 6b42        E...!F.....8?.kB
0x0010   d445 a235 d6d2 0050 15c1 f39e 3476 7bfb        .E.5...P....4v{.
0x0020   5018 40e8 f4b3 0000 4745 5420 2f73 6572        P.@.....GET./ser
0x0030   766c 6574 2f6b 616c 6973 682e 5365 6375        vlet/kalish.Secu
0x0040   7269 7479 2048 5454 502f 312e 310d 0a41        rity.HTTP/1.1..A
0x0050   6363                                          cc
```

For the sake of completeness, WHOIS information is included:

```
inetnum: 212.69.162.48 - 212.69.162.63
netname: ABOVENET-REITER
descr: Net of Abovenet Communications
GmbH / Austria
descr: with REITER country:
AT admin-c: THAL4-RIPE
tech-c: JH5258-RIPE
status: ASSIGNED PA
remarks: | http://www.abovenet.at/
remarks: | NETWORK OPERATIONS CENTER
remarks: | [phone]: +43 1 21122 1111
remarks: | [e-mail]: noc@abovenet.at
notify: juergen.hasenauer@rizit.at
mnt-by: MNT-AT-ABOVENET
changed: juergen.hasenauer@rizit.at
20001116
source: RIPE
```

```
person: Monika Thalinger
address: [organization]: AboveNet
Communications GmbH
address: [street address]:
Hollandstrasze 11-13
address: [postal code]: A-1020
address: [city]: Wien, Vienna
address: [country]: AT, Austria, Europe
phone: +43 1 2128644 5202
fax-no: +43 1 2128644 5225
e-mail: office@abovenet.at
nic-hdl: THAL4-RIPE
remarks: # http://www.abovenet.at/
remarks: # NETWORK OPERATIONS CENTER
remarks: # [phone]: +43 1 21122 1111
remarks: # [e-mail]: noc@abovenet.at
notify: pmandl@abovenet.at
```

Combining the suspect IP address with the tcpdump trace produces the following URL:
http://www.mycgiserver.com/servlet/kalish.Security



My experience confirms what the packet dumps show. After clicking on the link to start the Security servlet, I clicked the browser's "Stop" button (thereby sending a RST). Regardless, the Java servlet continued it's portscan.

Correlations:

Many of these personal security scanning services exist on the Internet. An Internet search on "Kalish Security Scanner" produced many hits:
http://kernel.org.in/mirrors/munitions/documents/linux-security-links
http://br.geocities.com/segurancamaxima/smtest.htm
http://www.dslreports.com/security/sec027.htm

Evidence of Active Targeting:
An internal user initiated this scan. No targeting was required.

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

Criticality	1	This is an unused host with no business purpose; it does not house critical data.
Lethality	1	Portscans are used to gather information about the host.
System Countermeasures	3	The system was a patched installation of Windows NT;
Network Countermeasures	1	The system is fully exposed to the Internet; no router or firewall filters traffic destined for the network.
Severity	**-2**	

Multiple Choice Test Question:
Which tcpdump flag displays packet information in hex?
A) –x
B) –h
C) –a
D) –X

Answer: A

Detect 2: wu-ftpd successful buffer overflow

```
 [**] FTP EXPLOIT format string [**]
11/10-13:17:25.568681 203.56.181.3:4828 -> MY.NET.107.197:21
TCP TTL:43 TOS:0x0 ID:0 IpLen:20 DgmLen:76 DF
***AP*** Seq: 0xF3F544F1  Ack: 0x8A2572AC  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 84660547 181229560
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

[**] FTP site exec [**]
11/10-13:17:26.085102 203.56.181.3:4828 -> MY.NET.107.197:21
TCP TTL:43 TOS:0x0 ID:0 IpLen:20 DgmLen:468 DF
***AP*** Seq: 0xF3F54509  Ack: 0x8A2572EA  Win: 0x16D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 84660598 181229611
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

```
repeated at:
11/10-13:17:26.618268 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:27.693407 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:28.227427 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:28.757166 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:29.286825 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:29.877275 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:30.276478 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:30.807948 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:31.347110 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:32.339073 203.56.181.3:4828 -> MY.NET.107.197:21
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

[**] FTP EXPLOIT wu-ftpd 2.6.0 site exec overflow [**]
11/10-13:17:32.879014 203.56.181.3:4828 -> MY.NET.107.197:21
TCP TTL:43 TOS:0x0 ID:0 IpLen:20 DgmLen:563 DF
***AP*** Seq: 0xF3F555D0  Ack: 0x8A258D5D  Win: 0x694C  TcpLen: 32
TCP Options (3) => NOP NOP TS: 84661278 181230289
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

repeated at:
11/10-13:17:33.420417 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:34.455568 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:35.000989 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:36.080331 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:36.619320 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:37.659582 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:38.199841 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:38.739175 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:39.280320 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:39.820092 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:40.805372 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:41.350282 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:41.889145 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:42.430033 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:42.970059 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:43.954771 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:44.499024 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:45.040051 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:45.579630 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:46.120009 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:46.669060 203.56.181.3:4828 -> MY.NET.107.197:21
11/10-13:17:47.225915 203.56.181.3:4828 -> MY.NET.107.197:21
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

[**] FTP site exec [**]
11/10-13:17:47.636032 203.56.181.3:4828 -> MY.NET.107.197:21
TCP TTL:43 TOS:0x0 ID:0 IpLen:20 DgmLen:520 DF
***AP*** Seq: 0xF3F58170  Ack: 0x8A25F14A  Win: 0x82D0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 84662753 181231765
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

[**] FTP site exec [**]
11/10-13:17:48.177830 203.56.181.3:4828 -> MY.NET.107.197:21
TCP TTL:43 TOS:0x0 ID:0 IpLen:20 DgmLen:563 DF
***AP*** Seq: 0xF3F58344  Ack: 0x8A25F5A6  Win: 0x82D0  TcpLen: 32
```

```
TCP Options (3) => NOP NOP TS: 84662808 181231819
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

[**] FTP EXPLOIT wu-ftpd 2.6.0 linux overflow [**]
11/10-13:17:49.941951 203.56.181.3:4828 -> MY.NET.107.197:21
TCP TTL:43 TOS:0x0 ID:0 IpLen:20 DgmLen:201 DF
***AP*** Seq: 0xF3F58543  Ack: 0x8A25F988  Win: 0x8FAA  TcpLen: 32
TCP Options (3) => NOP NOP TS: 84662985 181231895
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

Source of Trace:
The author's core external network.

Type of Event Generator:
SNORT v1.8.1 with slightly modified ruleset, running on NetBSD 1.5.2 i386 (Intel Pentium II), . The specific rules which produced the alert are as follows (from "ftp.rules"):

1) ftp.rules:alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP site exec"; content: "site exec"; nocase; flags: A+; reference:bugtraq,2241; reference:arachnids,317; classtype:bad-unknown; sid:361; rev:2;)

2) ftp.rules:alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP EXPLOIT wu-ftpd 2.6.0 site exec overflow"; content: "SITE EXEC %p"; nocase; flags: A+; depth: 16; reference:arachnids,285; classtype:attempted-admin; sid:345; rev:1;)

3) ftp.rules:alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP EXPLOIT wu-ftpd 2.6.0 site exec overflow"; content: "|66 25 2E 66 25 2E 66 25 2E 66 25 2E 66 25 2E|"; flags: A+; depth: 32; reference:arachnids,286; classtype:attempted-admin; sid:346; rev:1;)

4) ftp.rules:alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP EXPLOIT wu-ftpd 2.6.0 linux overflow"; content: "|31c031db 31c9b046 cd80 31c031db|"; flags: A+; reference:arachnids,287; classtype:attempted-admin; sid:344; rev:1;)

From compromised host "secure" log:
Nov 10 11:13:36 localhost in.ftpd[26672]: connect from 203.56.181.3
Nov 10 12:49:23 localhost in.ftpd[26693]: connect from 203.56.181.3

Probability the source address was spoofed:
Extremely Low. A full TCP connection must be established with the victim in order to deliver the attack.

TTL analysis is consistent. The TTL value for packets sent from the attacker is highlighted below:

```
13:17:24.427772 cliff.surfnetcity.com.au.4828 > MY.NET.107.197.ftp: S 4092937433
:4092937433(0) win 5840 <mss 1460,sackOK,timestamp 84660433 0,nop,wscale 0> (DF)
0x0000   4500 003c 0000 4000 2b06 23df cb38 b503        E..<..@.+.#..8..
0x0010   xxyy 6bc5 12dc 0015 f3f5 44d9 0000 0000        ?.k.......D.....
0x0020   a002 16d0 e3bc 0000 0204 05b4 0402 080a        ................
0x0030   050b d0d1 0000 0000 0103 0300                  ............
```

0x2B is "43" in decimal. This number, plus the number of network hops back to the attacker will approximate the initial TTL. The initial TTL value can help the analyst narrow down the attacking operating system. The traceroute is shown below:

```
traceroute to 203.56.181.3 (203.56.181.3), 30 hops max, 40 byte packets
 1  MY.NET.107.193 (MY.NET.107.193)  1.174 ms  4.220 ms  1.111 ms

[omitted]

14  0.SO-6-2-0.XR1.SYD4.ALTER.NET (210.80.51.74)  231.666 ms  239.614 ms  239.708 ms
15  so-0-0-1.XR2.MEL1.ALTER.NET (210.80.33.18)  247.456 ms  245.595 ms  247.642 ms
16  412.ATM8-0-0.GW1.MEL1.ALTER.NET (210.80.32.26)  244.683 ms  248.169 ms  247.177 ms
17  lavalink-mel1-gw.customer.alter.net (203.166.91.214)  255.708 ms  255.649 ms
255.629 ms
18  cliff.surfnetcity.com.au (203.56.181.3)  255.301 ms  255.973 ms  250.480 ms
```

18 hops plus a TTL of 43 results in an initial TTL of 61, which is close to the typical initial value of 64 for many versions of UNIX, including Linux. The attack tool is typically launched from a UNIX host, correlating this finding.

Description of Attack:
This attack exploits improper format string usage in vsnprint() to overwrite an arbitrary portion of stack space. This allows an attacker to run arbitrary operating system commands "by inserting string-formatting operators into command input, which are incorrectly parsed by the FTP server." (From http://www.securityfocus.com/advisories/2374) The offending snippet of code from the wu-ftpd-2.6.0 source code (from www.cs.berkeley.edu/~ushankar/research/percents/percents.pdf, which presents an interesting methodology for detecting such vulnerabilities) is shown below:

```
while (fgets(buf, sizeof buf, f))
{
      lreply(200, buf);
. . .
}

 void lreply(int n, char *fmt, ...)
{ . . .
      vsnprintf(buf, sizeof buf, fmt, ap);
. . .
}
```

Other descriptions of this attack:
http://www.securityfocus.com/advisories/2374
http://ciac.llnl.gov/ciac/bulletins/k-054.shtml
http://www.securityfocus.com/bid/2241
http://www.whitehats.com/info/IDS317

Attack Mechanism:
Several tools have been released into the public domain to facilitate this attack. The code snippet below is from "bobek.c", a modified version of wu2600.c. This function builds the exploit string

that gets passed to the ftp server. Line 9 shows the "%.f" signature which is one indicator of this attack.

```
1 for (i=eipoff;;i+=8)
2     {
3           fprintf(stderr, "at offset %d\n", i);
4           strcpy(sendbuf, "SITE EXEC ");
5
6           for (j=0;j<align;j++) strcat(sendbuf, "a");
7           strcat(sendbuf, "abcd");
8
9           for (j=0;j<eipoff/8;j++) strcat(sendbuf, "%%.f");
10          for (j=0;j<(i-eipoff)/8;j++) strcat(sendbuf, "%%d");
11          strcat(sendbuf, "|%%.8x|%%.8x");
```

Here is the resulting tcpdump capture:

```
13:17:46.120004 cliff.surfnetcity.com.au.4828 > MY.NET.107.197.ftp: P
14563:15074(511) ack 30377 win 33488 <nop,nop,timestamp 84662602 181231613>
(DF)
0x0000   4500 0233 0000 4000 2b06 21e8 cb38 b503        E..3..@.+.!..8..
0x0010   xxyy 6bc5 12dc 0015 f3f5 7dbc 8a25 e880        ?.k.......}..%..
0x0020   8018 82d0 45d5 0000 0101 080a 050b d94a        ....E..........J
0x0030   0acd 5ffd 5349 5445 2045 5845 4320 3720        .._.SITE.EXEC.7.
0x0040   acb2 ffff bf25 2e66 252e 6625 2e66 252e        .....%.f%.f%.f%.
0x0050   6625                                          f%
```

However, tcpdump captured other signatures for the same attack which would be flagged by the default SNORT rules as "FTP site exec" instead of a more serious remote root exploit.

```
13:17:47.636027 cliff.surfnetcity.com.au.4828 > MY.NET.107.197.ftp: P
15511:15979(468) ack 32627 win 33488 <nop,nop,timestamp 84662753 181231765>
(DF)
0x0000   4500 0208 0000 4000 2b06 2213 cb38 b503        E.....@.+."..8..
0x0010   xxyy 6bc5 12dc 0015 f3f5 8170 8a25 f14a        ?.k........p.%.J
0x0020   8018 82d0 4466 0000 0101 080a 050b d9e1        ....Df..........
0x0030   0acd 6095 5349 5445 2045 5845 4320 3720        ..`.SITE.EXEC.7.
0x0040   4141 4141 5073 5073 4241 4141 5073 5073        AAAAPsPsBAAAPsPs
0x0050   4341                                          CA

13:17:48.177825 cliff.surfnetcity.com.au.4828 > MY.NET.107.197.ftp: P
15979:16490(511) ack 33743 win 33488 <nop,nop,timestamp 84662808 181231819>
(DF)
0x0000   4500 0233 0000 4000 2b06 21e8 cb38 b503        E..3..@.+.!..8..
0x0010   xxyy 6bc5 12dc 0015 f3f5 8344 8a25 f5a6        ?.k........D.%..
0x0020   8018 82d0 35f1 0000 0101 080a 050b da18        ....5...........
0x0030   0acd 60cb 5349 5445 2045 5845 4320 3720        ..`.SITE.EXEC.7.
0x0040   54d0 ffff bf50 7350 7355 d0ff ffbf 5073        T....PsPsU....Ps
0x0050   5073                                          Ps
```

Correlations:

This network receives port 21 SYN scans approximately once per day. This exploit has been "in the wild" since October 1999, but was not made public until June 2000. FTP was the fourth

most attacked port on 13 November, according to Incidents.org. Interestingly, a small submission spike at incidents.org occurred on the same day the reviewed host was initially scanned (see below).

Report for Port # 21 - FTP

Date	Count	Percent of Submissions	
2001-11-13	37107	4.58%	
2001-11-12	24130	2.11%	
2001-11-11	10329	0.96%	
2001-11-10	109260	10.24%	
2001-11-09	13190	3.68%	

AusCERT Advisory:
ftp://ftp.auscert.org.au/pub/auscert/advisory/AA-2000.02

Posted by Jose Nazario on incidents.org:
http://www.incidents.org/archives/y2k/062800-1000.htm

John Johnston's post also to incidents.org
http://www.incidents.org/archives/y2k/062300-1430.htm

An analysis of a similar attack on one of the Honeynet Project Redhat servers:
http://project.honeynet.org/scans/scan19/scan/som28/analysis.html

Evidence of Active Targeting:
This host was SYN scanned approximately one hour before the attack (from SNORTs portscan.log). Only active IPs were scanned.

```
Date/Time       Source IP              Dest IP           Packet/TYPE
Nov 10 11:41:38 203.56.181.3:21     -> MY.NET.107.66:21  SYN ******S*
```

```
Nov 10 11:41:42 203.56.181.3:3222   -> MY.NET.107.194:21 SYN ******S*
Nov 10 11:41:38 203.56.181.3:21     -> MY.NET.107.195:21 SYN ******S*
Nov 10 11:41:38 203.56.181.3:21     -> MY.NET.107.196:21 SYN ******S*
Nov 10 11:41:39 203.56.181.3:3201   -> MY.NET.107.197:21 SYN ******S*
Nov 10 11:41:38 203.56.181.3:21     -> MY.NET.107.198:21 SYN ******S*
Nov 10 11:41:38 203.56.181.3:21     -> MY.NET.107.200:21 SYN ******S*
```

Severity:

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

Criticality	1	This is an unused host with no business purpose; it does not house critical data.
Lethality	5	The exploit gained root access.
System Countermeasures	1	The system was a default installation of Linux; no patches were applied.
Network router Countermeasures	1	The system is fully exposed to the Internet; no border or firewall filters traffic destined for the network.
Severity	**4**	

Defensive Recommendations:
Since the system serves no useful purpose, removing it from service and disconnecting it from the Internet is best. If the system must remain on the Internet, more aggressive steps must be taken to properly install and maintain the system.

Additionally, another SNORT rule would catch the other wu-ftpd exploit code:

alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP EXPLOIT wu-ftpd 2.6.0 site exec overflow"; content: "|50 73 50 73|"; flags: A+; depth: 32; reference:arachnids,286; classtype:attempted-admin; sid:346; rev:1;)

Multiple Choice Test Question:
Given the packet dump:

```
13:17:24.427772 cliff.surfnetcity.com.au.4828 > MY.NET.107.197.ftp: S 4092937433
:4092937433(0) win 5840 <mss 1460,sackOK,timestamp 84660433 0,nop,wscale 0> (DF)
0x0000   4500 003c 0000 4000 2b06 23df cb38 b503
0x0010   xxyy 6bc5 12dc 0015 f3f5 44d9 0000 0000
0x0020   a002 16d0 e3bc 0000 0204 05b4 0402 080a
0x0030   050b d0d1 0000 0000 0103 0300
```

Which byte of the IP header tells us the TTL?
A) 7 (40)
B) 8 (00)
C) 9 (2b)
D) 10 (06)

Answer: C

Detect #3: ICMP Superscan Echo

```
 [**] [1:474:1] ICMP superscan echo [**]
[Classification: Attempted Information Leak] [Priority: 3]
11/24-00:51:37.881707 66.1.247.137 -> MY.NET.107.66
ICMP TTL:111 TOS:0x0 ID:50667 IpLen:20 DgmLen:36
Type:8  Code:0  ID:2   Seq:47353  ECHO

[**] [1:474:1] ICMP superscan echo [**]
[Classification: Attempted Information Leak] [Priority: 3]
11/24-00:51:42.770376 66.1.247.137 -> MY.NET.107.196
ICMP TTL:111 TOS:0x0 ID:51744 IpLen:20 DgmLen:36
Type:8  Code:0  ID:2   Seq:48337  ECHO

[**] [1:474:1] ICMP superscan echo [**]
[Classification: Attempted Information Leak] [Priority: 3]
11/24-00:51:42.796367 66.1.247.137 -> MY.NET.107.197
ICMP TTL:111 TOS:0x0 ID:51749 IpLen:20 DgmLen:36
Type:8  Code:0  ID:2   Seq:48342  ECHO
TCP Options (5) => WS: 10 NOP MSS: 265 TS: 1061109567 0 EOL

 [**] [1:474:1] ICMP superscan echo [**]
[Classification: Attempted Information Leak] [Priority: 3]
11/24-00:51:42.816846 66.1.247.137 -> MY.NET.107.198
ICMP TTL:111 TOS:0x0 ID:51753 IpLen:20 DgmLen:36
Type:8  Code:0  ID:2   Seq:48346  ECHO
```

Source of Trace:
The author's core external network.

Type of Event Generator:
SNORT v1.8.1 with slightly modified ruleset, running on NetBSD 1.5.2 i386 (Intel Pentium II), attached to 384Kb leased line. The specific rule which produced the alert (from "icmp.rules"):

alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP superscan echo"; content:"|0000000000000000|"; itype: 8; dsize:8; classtype:attempted-recon; sid:474; rev:1;)

```
00:51:37.881702 cpe-66-1-247-137.co.sprintbbd.net > MY.NET.107.66: icmp: echo
request
0x0000   4500 0024 c5eb 0000 6f01 a144 4201 f789        E..$....o..DB...
0x0010   xxyy 6b42 0800 fc46 0200 f9b8 0000 0000        ?.kB...F........
0x0020   0000 0000 0000 0000 0000 0000 0000             ..............
```

01 Protocol: ICMP
08 Echo Request

Probability the source address was spoofed:
Low. When the attacker received an ICMP echo reply, an immediate attempt was made to connect to the FTP service from the same source IP address.

```
traceroute to 66.1.247.137 (66.1.247.137), 30 hops max, 40 byte packets
 1  MY.NET.107.193 (MY.NET.107.193)  1.136 ms  1.130 ms  1.086 ms

[omitted]

14  car0102-vlan-1.den03.inflow.net (216.183.96.12)  90.465 ms  90.322 ms
90.094 ms
15  216.183.97.30 (216.183.97.30)  90.095 ms  90.099 ms  90.488 ms
16  24.221.208.202 (24.221.208.202)  94.157 ms  94.122 ms  94.437 ms
17  24.221.31.38 (24.221.31.38)  98.000 ms  95.703 ms  94.977 ms
18  cpe-66-1-247-137.co.sprintbbd.net (66.1.247.137)  155.166 ms  330.283 ms
690.316 ms

Sprint BWG (NETBLK-SPRINTBWG-1BL-COCS-3) SPRINTBWG-1BL-COCS-3
                                                    66.1.240.0 - 66.1.247.255
```

Description of Attack:
The attacker first selectively sent ICMP echo requests to only active hosts. The small time deltas between requests indicate this is an automated scan.

```
00:51:42.796363 cpe-66-1-247-137.co.sprintbbd.net > MY.NET.107.197: icmp:
echo request
00:51:42.798201 MY.NET.107.197 > cpe-66-1-247-137.co.sprintbbd.net: icmp:
echo reply (DF)
00:51:42.816842 cpe-66-1-247-137.co.sprintbbd.net > MY.NET.107.198: icmp:
echo request
00:51:42.816893 MY.NET.107.198 > cpe-66-1-247-137.co.sprintbbd.net: icmp:
echo reply
00:51:42.816978 MY.NET.107.198 > cpe-66-1-247-137.co.sprintbbd.net: icmp:
echo reply
```

When an ICMP echo reply is received, a connection to port 23/tcp (FTP) is attempted.

```
00:51:43.570882 cpe-66-1-247-137.co.sprintbbd.net.2707 > MY.NET.107.197.ftp:
S 3856185774:3856185774(0) win 17520 <mss 1460,nop,nop,sackOK> (DF)
00:51:43.632478 cpe-66-1-247-137.co.sprintbbd.net.2709 > MY.NET.107.198.ftp:
S 3856328425:3856328425(0) win 17520 <mss 1460,nop,nop,sackOK> (DF)
```

Attack Mechanism:
The pattern above is consistent with Superscan. The tool has the ability to "connect to any discovered open port using user-specified "helper" applications and assign a custom helper application to any port." (http://www.hot.ee/hagelberg/Hacks.htm). In addition, the IP ID is always set to 0x0200. The trace below shows what the scanning tool discovered.

```
00:51:43.709012 MY.NET.107.197.ftp > cpe-66-1-247-137.co.sprintbbd.net.2707: . ack 3
win 8760 (DF)
0x0000   4500 0028 663f 4000 ff06 3064 xxyy 6bc5        E..(f?@...0d?.k.
0x0010   4201 f789 0015 0a93 c462 797f e5d8 b9b1        B........by.....
0x0020   5010 2238 c05b 0000 5555 5555 5555             P."8.[..UUUUUU
00:51:43.730894 MY.NET.107.197.ftp > cpe-66-1-247-137.co.sprintbbd.net.2707: P
1:43(42) ack 3 win 8760 (DF)
0x0000   4500 0052 6640 4000 ff06 3039 xxyy 6bc5        E..Rf@@...09?.k.
0x0010   4201 f789 0015 0a93 c462 797f e5d8 b9b1        B........by.....
```

```
0x0020   5018 2238 72fa 0000 3232 3020 6d69 6e69        P."8r...220.mini
0x0030   6d65 2046 5450 2073 6572 7665 7220 2853        me.FTP.server.(S
0x0040   756e 4f53 2035 2e36 2920 7265 6164 792e        unOS.5.6).ready.
0x0050   0d0a                                           ..
00:51:43.888319 cpe-66-1-247-137.co.sprintbbd.net.2707 > MY.NET.107.197.ftp: F 3:3(0)
ack 43 win 17478 (DF)
0x0000   4500 0028 cb68 4000 6f06 5b3b 4201 f789        E..(.h@.o.[;B...
0x0010   xxyy 6bc5 0a93 0015 e5d8 b9b1 c462 79a9        ?.k.........by.
0x0020   5011 4446 9e22 0000 0000 0000 0000             P.DF."........
00:51:43.888574 MY.NET.107.197.ftp > cpe-66-1-247-137.co.sprintbbd.net.2707: P
43:76(33) ack 4 win 8760 (DF)
0x0000   4500 0049 6641 4000 ff06 3041 xxyy 6bc5        E..IfA@...0A?.k.
0x0010   4201 f789 0015 0a93 c462 79a9 e5d8 b9b2        B........by.....
0x0020   5018 2238 184c 0000 3530 3020 2727 3a20        P."8.L..500.'':.
0x0030   636f 6d6d 616e 6420 6e6f 7420 756e 6465        command.not.unde
0x0040   7273 746f 6f64 2e0d 0a                         rstood...
00:51:43.888647 MY.NET.107.197.ftp > cpe-66-1-247-137.co.sprintbbd.net.2707: . ack 4
win 8760 (DF)
0x0000   4500 0028 6642 4000 ff06 3061 xxyy 6bc5        E..(fB@...0a?.k.
0x0010   4201 f789 0015 0a93 c462 79ca e5d8 b9b2        B........by.....
0x0020   5010 2238 c00f 0000 5555 5555 5555             P."8....UUUUUU
```

It is unknown what the attacker attempted to pass to the FTP server.

Correlations:
This GCIA-compliant write-up correlates Superscan activity and similar FTP queries:
http://www.ini2.net/mel/snort_trace/proxy-3110.html
Another reference showing Superscan alerts around the end of July 2001:
http://www.incidents.org/archives/intrusions/msg01221.html
Recommendation to improve Superscan detects:
http://archives.neohapsis.com/archives/snort/2001-02/0073.html
The superscan tool was purchased by Foundstone from Robin Keir (who is now an employee):
http://www.foundstone.com/rdlabs/termsofuse.php?filename=superscan.exe

Evidence of Active Targeting:
This attacker has targeted this network range before. ICMP echo requests are only sent to IP addresses which were recently active.

```
00:51:37.881702 cpe-66-1-247-137.co.sprintbbd.net > MY.NET.107.66: icmp: echo request
00:51:39.338546 cpe-66-1-247-137.co.sprintbbd.net > MY.NET.107.100: icmp: echo request
00:51:42.770371 cpe-66-1-247-137.co.sprintbbd.net > MY.NET.107.196: icmp: echo request
00:51:42.796363 cpe-66-1-247-137.co.sprintbbd.net > MY.NET.107.197: icmp: echo request
00:51:42.798201 MY.NET.107.197 > cpe-66-1-247-137.co.sprintbbd.net: icmp: echo reply
(DF)
00:51:42.816842 cpe-66-1-247-137.co.sprintbbd.net > MY.NET.107.198: icmp: echo request
00:51:42.816893 MY.NET.107.198 > cpe-66-1-247-137.co.sprintbbd.net: icmp: echo reply
00:51:42.816978 MY.NET.107.198 > cpe-66-1-247-137.co.sprintbbd.net: icmp: echo reply
```

Severity:

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

Criticality	1	This is an unused host with no business purpose; it does not house critical data.

Lethality	3	This scan is to gather information. However, the attacker also queried the FTP server for software version and attempted to connect.
System Countermeasures	1	The system was a default installation of Solaris 2.6; no patches were applied.
Network Countermeasures	1	The system is fully exposed to the Internet; no router or firewall filters traffic destined for the network.
Severity	**2**	

Defensive Recommendations:
Determine victim's business purpose; if it is not required to face the Internet, remove it. If required to remain and services which use inetd must also remain available, implement tcpwrappers. Ensure system is actively maintained with current operating system updates. Require a minimum baseline standard configuration for all computers which must attach to the Internet.

Multiple Choice Test Question:
Given the hex packet dump

```
0x0000   4500 0024 c5eb 0000 6f01 a144 4201 f789
0x0010   xxyy 6b42 0800 fc46 0200 f9b8 0000 0000
0x0020   0000 0000 0000 0000 0000 0000 0000
```

What kind of ICMP packet is this?
A) ECHO request
B) ECHO reply
C) Address mask request
D) Address mask reply

Answer: A

Detect #4: RPC Portmap request rstatd

Rstatd is an RPC service which allows a remote user to collect performance information. Unfortunately, like most other RPC services, it has several vulnerabilities which could allow an attacker to gain root-level access remotely.

```
 [**] RPC portmap request rstatd [**]
11/25-11:10:21.072726 61.33.33.124:1023 -> MY.NET.107.197:111
UDP TTL:43 TOS:0x0 ID:22939 IpLen:20 DgmLen:84
Len: 64
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

Source of Trace:
The author's core external network.

Type of Event Generator:
SNORT v1.8.1 with slightly modified ruleset, running on NetBSD 1.5.2 i386 (Intel Pentium II), attached to 384Kb leased line. The specific rule which produced the alert (from "rpc.rules"):
alert udp $EXTERNAL_NET any - $HOME_NET 111 (msg:"RPC portmap request rstatd"; content: "|01 86 A0 00 00|"; reference:arachnids,10;)

Probability the source address was spoofed:
Low. The attacker first scanned for port 111/tcp (portmapper), then queried it for the rstatd service, which happened to reside on port 32772/udp.

Description of Attack:
This event is an information gathering activity. No exploit was attempted. The attacker first scans target hosts for port 111/tcp, the queries the portmapper for the location of the rstatd service.

Attack Mechanism:
The target host begins by letting the attacker know that the portmapper service is listening. (An excellent RPC reference can be found http://anguilla.u.arizona.edu/doc_link/en_US/a_doc_lib/aixprggd/progcomc/rpc_msg.htm)

```
11:10:20.818880 MY.NET.107.197.sunrpc > 61.33.33.124.2494: S
1735423483:1735423483(0) ack 2905742412 win 10136 <nop,nop,timestamp 42127363
178720590,nop,wscale 0,mss 1460> (DF)
                         4500 003c 201f 4000 ff06 515e xxyy 6bc5
                         3d21 217c 006f 09be 6770 75fb ad32 1c4c
                         a012 2798 7b8f 0000 0101 080a 0282 d003
                         0aa7 0f4e 0103 0300 0204 05b4
```

A three-way handshake is completed.

```
11:10:21.072233 61.33.33.124.2494 > MY.NET.107.197.sunrpc: . ack 1 win 32120
<nop,nop,timestamp 178720616 42127363> (DF)
```

```
                        4500 0034 599a 4000 2b06 ebeb 3d21 217c
                        xxyy 6bc5 09be 006f ad32 1c4c 6770 75fc
                        8010 7d78 5159 0000 0101 080a 0aa7 0f68
                        0282 d003
```

A request is made

```
11:10:21.072721 61.33.33.124.1023 > MY.NET.107.197.sunrpc:  udp 56
                        4500 0054 599b 0000 2b11 2bc0 3d21 217c
                        xxyy 6bc5 03ff 006f 0040 4029 4d9c 5588
                        0000 0000 0000 0002 0001 86a0 0000 0002
                        0000 0003 0000 0000 0000 0000 0000 0000
                        0000 0000 0001 86b8 0000 0001 0000 0011
                        0000
```

```
11    Protocol: UDP
40    Length: 64                    0001 86a0  RPC Program
4d9c 5588  XID                     0000 0002  Program Version
0000 0000  Message Type (Call)     0000 0003  Procedure Number
0000 0002  RPC Version             0000 0003  Authentication
```

```
11:10:21.074798 MY.NET.107.197.sunrpc > 61.33.33.124.1023:  udp 28 (DF)
                        4500 0038 2020 4000 ff11 5156 xxyy 6bc5
                        3d21 217c 006f 03ff 0024 cdcf 4d9c 5588
                        0000 0001 0000 0000 0000 0000 0000 0000
                        0000 0000 0000 8004
```

```
4d9c 5588  XID
0000 0001  RPC Reply
0000 0000  Message Accepted
```

```
11:10:21.280562 61.33.33.124.600 > MY.NET.107.197.32772:  udp 1076
                        4500 0450 599d 0000 2b11 27c2 3d21 217c
                        xxyy 6bc5 0258 8004 043c 8c69 4b90 04d2
                        0000 0000 0000 0002 0001 86b8 0000 0001
                        0000 0001 0000 0001 0000 0020 3c01 2590
                        0000 0009 6c6f 6361 6c68 6f73 7400 0000
                        0000
```

```
4b90 04d2  XID              0000 0001  Program Version
0000 0000  RPC Call         0000 0001  Procedure #
0000 0002  RPC Version      0000 0001  Authentication
0001 86b8  RPC Program
```

```
11:10:21.281221 MY.NET.107.197.32772 > 61.33.33.124.600:  udp 32 (DF)
                        4500 003c 2021 4000 ff11 5151 xxyy 6bc5
                        3d21 217c 8004 0258 0028 229d 4b90 04d2
                        0000 0001 0000 0000 0000 0000 0000 0000
                        0000 0000 0000 0000 0000 0003
```

```
4b90 04d2  XID
0000 0001  RPC Reply
0000 0000  Message Accepted
```

```
11:10:21.536308 61.33.33.124.2494 > MY.NET.107.197.sunrpc: F 1:1(0) ack 1 win
32120 <nop,nop,timestamp 178720667 42127363> (DF)
                         4500 0034 599e 4000 2b06 ebe7 3d21 217c
                         xxyy 6bc5 09be 006f ad32 1c4c 6770 75fc
                         8011 7d78 5125 0000 0101 080a 0aa7 0f9b
                         0282 d003
```

The connection termination is begun

Suspect host WHOIS information:

```
inetnum:      61.33.33.0 - 61.33.33.127
netname:      WAWOO55561D
descr:        Wawoo Free Communication
country:      KR
admin-c:      HK72-AP
tech-c:       DB50-AP
notify:       b0055561@users.bora.net
mnt-by:       MAINT-KR-DACOM
changed:      b0055561@users.bora.net 20001031
source:       APNIC

role:         DACOM BORANET
address:      DACOM Bldg., 706-1, Yoeksam-dong, Kangnam-ku, Seoul
phone:        +82-2-6220-7755
fax-no:       +82-2-6220-0706
e-mail:       ipadm@nic.bora.net
```

Correlations:
CVE-2000-0666 describes the linux rpc.rstatd remote root exploit (also used by the Ramen Worm). No correlation could be made for this netblock with incidents.org.
This note describes a false positive alert: http://archives.neohapsis.com/archives/incidents/2001-06/0045.html

Evidence of Active Targeting:
Specific IP addresses were targeted by this scan, indicating the attacker had prior knowledge of the networked hosts. The trace detailing the portscan demonstrates this analysis.

```
11:10:18.640518 61.33.33.124.2363 > MY.NET.107.66.sunrpc: S
2900975299:2900975299(0) win 32120 <mss 1460,sackOK,timestamp 178720373
0,nop,wscale 0> (DF)
11:10:18.751012 61.33.33.124.2397 > MY.NET.107.100.sunrpc: S
2896295420:2896295420(0) win 32120 <mss 1460,sackOK,timestamp 178720388
0,nop,wscale 0> (DF)
11:10:20.818294 61.33.33.124.2494 > MY.NET.107.197.sunrpc: S
2905742411:2905742411(0) win 32120 <mss 1460,sackOK,timestamp 178720590
0,nop,wscale 0> (DF)
```

Severity:

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

Criticality	1	This is an unused host with no business purpose; it does not house critical data.
Lethality	3	This scan is to gather information. However, the attacker also queried the portmapper for RPC information and
System Countermeasures	1	The system was a default installation of Solaris 2.6; no patches were applied.
Network Countermeasures	1	The system is fully exposed to the Internet; no router or firewall filters traffic destined for the network.
Severity	**2**	

Defensive Recommendations:
Except for not putting a server on the Internet, properly securing, patching and actively maintaining any Internet connected system is the best way to avoid compromise. Many guides are available which detail proper system administration.

- http://www.sans.org/infosecFAQ/unix/sec_solaris.htm
- http://www.serverworldmagazine.com/sunserver/2000/11/attack.shtml
- http://www.yassp.org

Multiple Choice Test Question:
Given the following RPC packet hex dump:

```
4500 0450 599d 0000 2b11 27c2 3d21 217c
xxyy 6bc5 0258 8004 043c 8c69 4b90 04d2
0000 0000 0000 0002 0001 86b8 0000 0001
0000 0001 0000 0001 0000 0020 3c01 2590
0000 0009 6c6f 6361 6c68 6f73 7400 0000
0000
```

What hex number is the RPC Program Number?
A) 4b90 04d2
B) 0000 0002
C) 0001 86b8
D) 6c6f 6361

Answer: C

Detect #5: FTP file system access

```
Nov 26 11:40:46 hostl proftpd[25707] hostl (cc56658-
a.emmen1.dr.nl.home.com[212.204.179.110]): FTP session opened.
Nov 26 11:40:47 hostl proftpd[25707] hostl (cc56658-
a.emmen1.dr.nl.home.com[212.204.179.110]): ANON anonymous: Login successful.
cc56658-a.emmen1.dr.nl.home.com UNKNOWN ftp [26/Nov/2001:11:40:47 -0500] "CWD
/pub/" 250 -
cc56658-a.emmen1.dr.nl.home.com UNKNOWN ftp [26/Nov/2001:11:40:47 -0500] "CWD
/public/" 550 -
cc56658-a.emmen1.dr.nl.home.com UNKNOWN ftp [26/Nov/2001:11:40:47 -0500] "MKD
011126174035p" 550 -
```

```
cc56658-a.emmen1.dr.nl.home.com UNKNOWN ftp [26/Nov/2001:11:40:47 -0500]
"PASS Wgpuser@home.com" 230 -
cc56658-a.emmen1.dr.nl.home.com UNKNOWN ftp [26/Nov/2001:11:40:48 -0500] "CWD
/pub/incoming/" 550 -
cc56658-a.emmen1.dr.nl.home.com UNKNOWN ftp [26/Nov/2001:11:40:49 -0500] "CWD
/incoming/" 250 -
cc56658-a.emmen1.dr.nl.home.com UNKNOWN ftp [26/Nov/2001:11:40:49 -0500] "MKD
011126174037p" 550 -
cc56658-a.emmen1.dr.nl.home.com UNKNOWN ftp [26/Nov/2001:11:40:50 -0500] "CWD
/" 250 -
cc56658-a.emmen1.dr.nl.home.com UNKNOWN ftp [26/Nov/2001:11:40:50 -0500] "CWD
/_vti_pvt/" 550 - cc56658-a.emmen1.dr.nl.home.com UNKNOWN ftp
[26/Nov/2001:11:40:50 -0500] "CWD /upload/" 550 -
cc56658-a.emmen1.dr.nl.home.com UNKNOWN ftp [26/Nov/2001:11:40:50 -0500] "MKD
011126174038p" 550 -
Nov 26 11:40:50 hostl proftpd[25707] hostl (cc56658-
a.emmen1.dr.nl.home.com[212.204.179.110]): FTP session closed.
```

Source of Trace:
A submission (http://www.incidents.org/archives/intrusions/msg02598.html) from Laurie Zirkle to incidents.org.

Type of Event Generator:
These appear to be system logs and FTP server logs.

Probability the source address was spoofed:
In order to take advantage of the information that was gathered during this event, the attacker would have had to established a full three-way handshake.

Description of Attack:
The automated tool appears to attempt to login to FTP servers anonymously.

```
Nov 26 11:40:46 hostl proftpd[25707] hostl (cc56658-
a.emmen1.dr.nl.home.com[212.204.179.110]): FTP session opened.
Nov 26 11:40:47 hostl proftpd[25707] hostl (cc56658-
a.emmen1.dr.nl.home.com[212.204.179.110]): ANON anonymous: Login successful.
```

If successful, the tool attempts several obviously suspicious maneuvers to enumerate potential vulnerabilities.

Attempted Action – Status Code	Meaning
"CWD /public/" 550	Change working directory; Requested action not taken. File unavailable
"CWD /pub/" 250	Change working directory; Requested file action okay, completed
"MKD 011126174035p" 550	Make directory; Requested action not taken. File unavailable
"PASS Wgpuser@home.com" 230	Enter password; User logged in, proceed.
"CWD /pub/incoming/" 550	Change working directory; Requested action not taken. File unavailable

"CWD /incoming/" 250	Change working directory; Requested file action okay, completed
"MKD 011126174037p" 550	Make directory; Requested action not taken. File unavailable
"CWD /" 250	Change working directory; Requested action not taken. File unavailable
"CWD /_vti_pvt/" 550	Change working directory; Requested action not taken. File unavailable
"MKD 011126174038p" 550	Make directory; Requested action not taken. File unavailable

The attacker was able to login anonymously, and change directories to /pub and /incoming. Attempts to create a remove directory failed.

Attack Mechanism:
This attack is definitely automated. The entire process described above took less than three seconds to complete. The purpose of the attack is to discover anonymous FTP servers with writeable directories. Once found, the FTP server usually becomes home to warez and porn.

Correlations:
David Allardyce (http://www1.dshield.org/pipermail/dshield/2001-October/001668.html) noticed a correlation with the anonymous password above (wgpuser@home.com).

Evidence of Active Targeting:
It appears several other hosts were targets of this attack.

```
Nov 26 11:40:46 hostt ftpd[29752]: refused connect from cc56658-
a.emmen1.dr.nl.home.com
Nov 26 11:40:47 hostz ftpd[16187]: refused connect from cc56658-
a.emmen1.dr.nl.home.com
Nov 26 11:40:47 hostsa ftpd[19375]: refused connect from cc56658-
a.emmen1.dr.nl.home.com
Nov 26 11:41:36 hostca in.ftpd[22381]: refused connect from cc56658-
a.emmen1.dr.nl.home.com
 a.emmen1.dr.nl.home.com
Nov 26 11:45:44 hostmau Connection attempt to TCP z.y.w.12:21 from
212.204.179.110:2967
```

Severity:

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

Criticality	1	Unkonwn; this is a known anonymous server
Lethality	3	This scan is to gather information. However, the attacker also queried the FTP server for software version and attempted to make changes to the file system.
System	2	The FTP server correctly limited the attacker's activities

Countermeasures		.
Network	1	Uknown, but the system appears fully exposed to the Internet.
Severity	**1**	

Defensive Recommendations:
Disable anonymous FTP unless absolutely necessary; use tcpwrappers if possible; continue to monitor system and server logs

Multiple Choice Test Question:
When viewing FTP server activity logs, it is important to understand various status codes in order to discover potentially malicious activity. Which activity code means "Requested file action okay, completed"
A) 230
B) 550
C) 250
D) 400

Answser: C

The University SNORT IDS Log Analysis

10 – 14 November 2001

Executive Summary

The University requested an analysis of five consecutive days worth of intrusion detection sensor data. Examination of the relevant logs has highlighted some areas which the University should be aware:

- The University's networks are "open": The number of hosts and network services accessible to the Internet seem to be large. Unless required for business reasons, access to these hosts should be drastically reduced.
- No minimum baseline standard: Hosts connected to the Internet seem to be generating many events across a range of network services. Properly secured and maintained hosts would eliminate these unnecessary alerts.
- Acceptable Use: the University's networks are host to many different kinds of (possibly) inappropriate network traffic such as Internet gaming, chat, IRC, and file sharing programs. The University should ensure that use of this type of software is consistent with University policy.

Objective and Scope

The objective of this report is to clearly communicate potential risks within The University's monitored network. "Scan", "Out-of-Spec" (OOS), and "Alert" logs collected from a SNORT intrusion detection system (IDS), generated between 10 November 2001 and 14 November 2001, inclusive, are considered.

Approach

The most basic, atomic elements for intrusion detection analysis are called "events of interest" (EOI). EOI are generated by tools designed to create audit data, and EOI are usually created in response to a stimulus. This report considers EOI generated by The University's SNORT external network IDS sensor, which creates EOI in response to specific network conditions. This particular data set includes 6,120,847 total events (5,053,685 scans, 1,066,070 alerts, 1,092 OOS). "Scan" events were tallied by source and destination IP address, and source and destination port. The 1,066,070 "alert" events were divided among 153 unique event types. All alert events were tallied by day, then separated into several logical categories:

- **Reconnaissance:** in order to attack a given network, an attacker must first collect information about it. This leaves traces on a network, some of which will cause SNORT to produce an alert. This is the first step preceding a network-based attack. Alert logs describe general port scans, while Scan and OOS logs provide detail such as source and destination host and packet data.
- **Exploitation:** these alert types signal an attempt to exploit a specific vulnerability. Attacks are correlated with scan and OOS data to determine stimulus and targeting activity.
- **Evidence of compromised hosts:** typical alerts include virus and trojan activity, reserved port usage, and inappropriate service usage.
- **Watchlists:** these are events which warrant special attention. Also included are custom alerts designed by the IDS sensor operator, and not part of the standard SNORT installation. Any occurrence must be investigated.
- **Denial of Service (DOS):** this type includes obvious DOS traffic or distributed DOS (DDOS) client/server interaction.

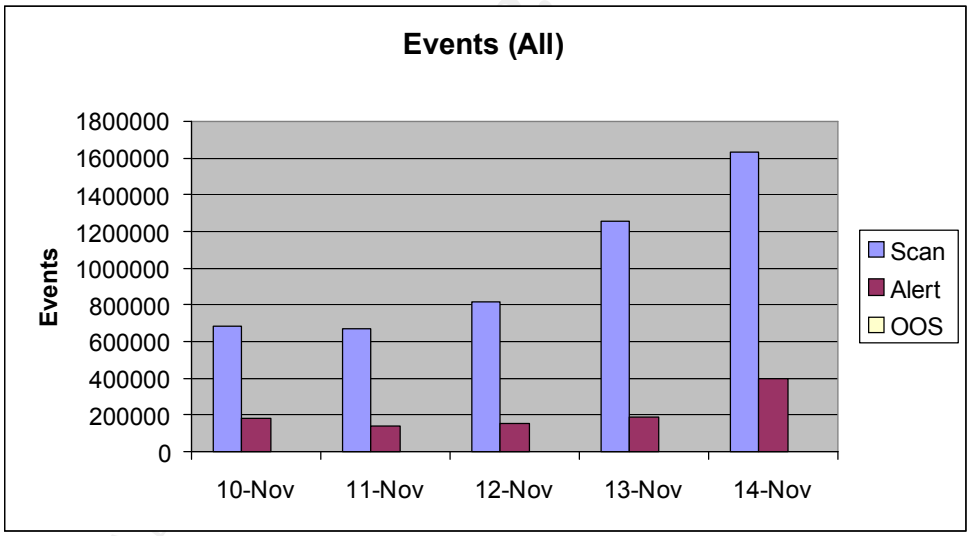
- **Suspect:** all other events are classified as "suspect." These events may or may not be malicious, but may require further investigation by the IDS operator or University network administrators.

All events were initially examined from a fairly high level. However, the events in each category were prioritized for further analysis by 1) examining the highest occurring events, 2) examining the highest potential risk events (since some high risk events may not occur very often), OR 3) examining associated hosts, or "top talkers". In addition, several "top ten" lists were created to provide a better basis for analysis, and possibly provide correlation for other alerts.

Finally, each investigated alert type or host was given a potential risk rating and defensive recommendation. These are intended to give The University network maintainers a basis to prioritize response activities.

Findings

Overview



Graph 1: All Events by Day

This graph indicates an overall increasing level of event traffic during the assessed time period.

Reconnaissance

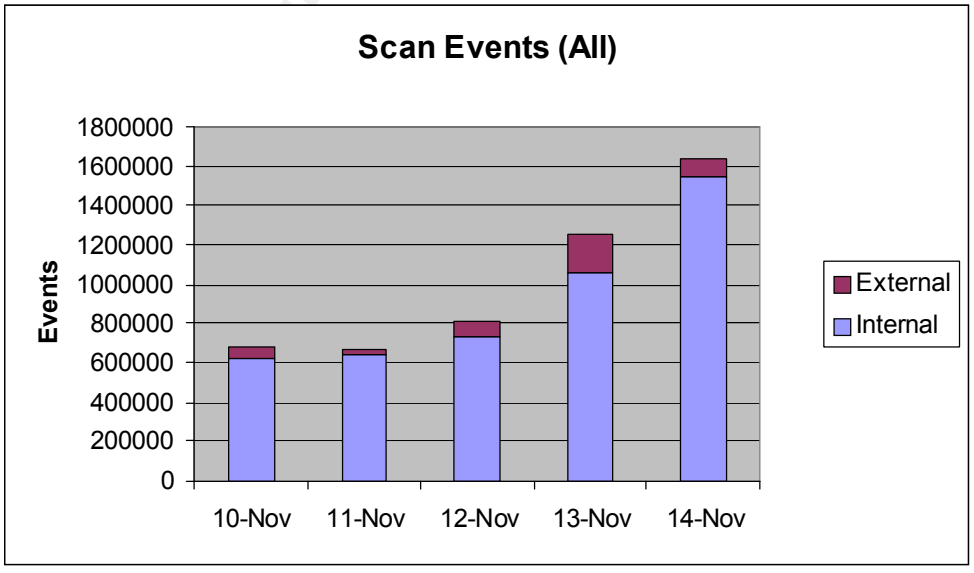
An attacker must first gather information about a target network and it's hosts before launching an attack. One way to do this is to perform "portscans." These scans enumerate listening services. SNORT scan event logs contain source and destination IP and port numbers for portscan alert events. Tallies of source IP addresses were taken from these logs in order to understand the most active scanning hosts. Results are listed below (percentages indicate fraction of scan events):

All Hosts (90.6%)	*Internal Hosts (96.7%)*	*External Hosts (89%)*
MY.NET.5.75	MY.NET.5.75	202.96.127.34
MY.NET.5.76	MY.NET.5.76	205.188.233.185
MY.NET.160.114	MY.NET.160.114	205.188.244.121
202.96.127.34	MY.NET.16.42	205.188.233.121
MY.NET.16.42	MY.NET.150.225	205.188.244.57
205.188.233.185	MY.NET.150.220	205.188.233.153
MY.NET.150.225	MY.NET.150.246	212.68.218.130
205.188.244.121	MY.NET.100.230	205.188.246.121
205.188.233.121	MY.NET.98.117	64.124.157.16
205.188.244.57	MY.NET.150.9	213.118.78.101

Table 1: Top Ten Scanning Hosts

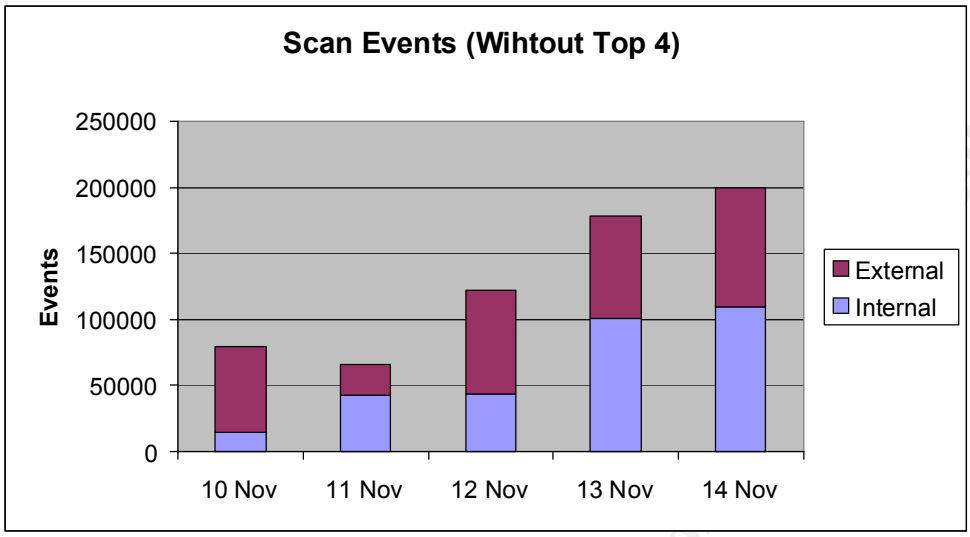
5,053,685 total scan events were recorded. Over 90% of all scan events were generated by the hosts in Table 1. Graph 1 (below) illustrates the daily breakdown of scan events by external and internal hosts.

Note: It is possible for IP addresses to be forged, or "spoofed." Therefore, any "top talkers" list is immediately suspect since it could have been influenced by many things, such as a large nmap scan using forged source IP addresses. In this case, the analyst is approaching the event logs as records of malicious intent until it is possible to make a determination about the authenticity of IP addresses. These tallies are based upon the Scan alert logs which were generated by SNORT. However, these events do give an indication of overall network traffic levels and make it possible to perform general trend analysis.



Graph 2: Daily Scan Events by Source

This graph shows the large majority of scan events were generated by internal hosts. In fact, 87.2% of all scan alerts were generated by the top four source hosts: MY.NET.5.75, MY.NET.5.76, MY.NET.160.114, 202.96.127.34. Removing the top four source hosts from the tallies produces a more normalized graph:

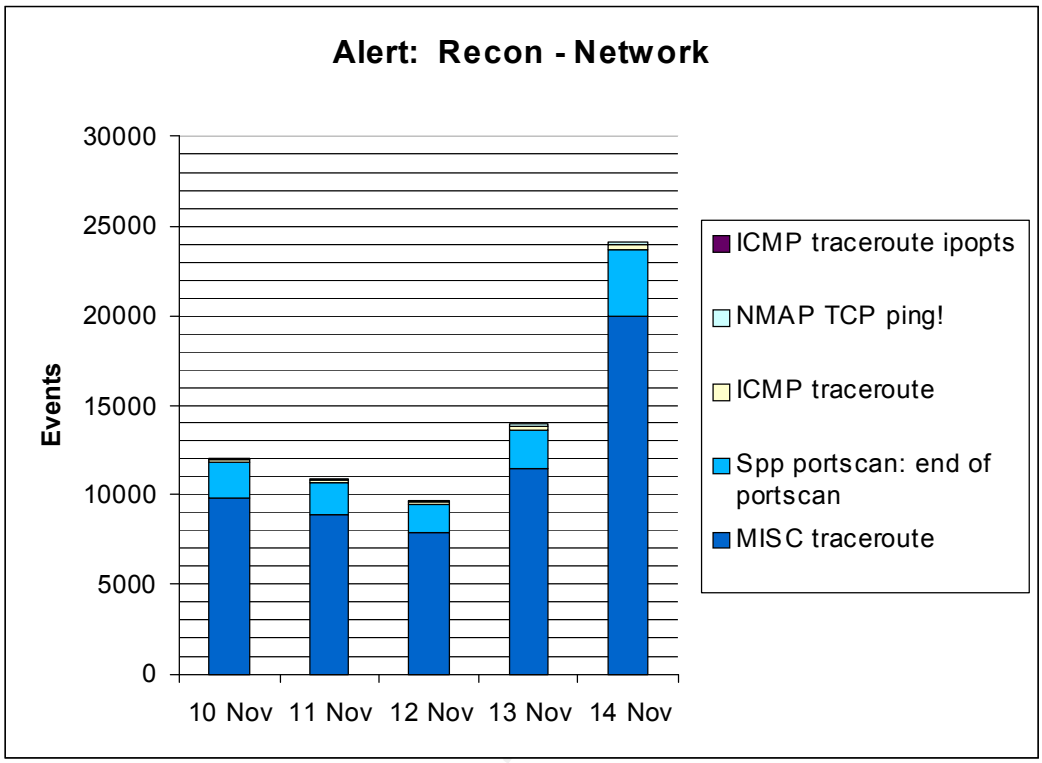


Graph 3: Daily Scan Events by Source (Top 4 removed)

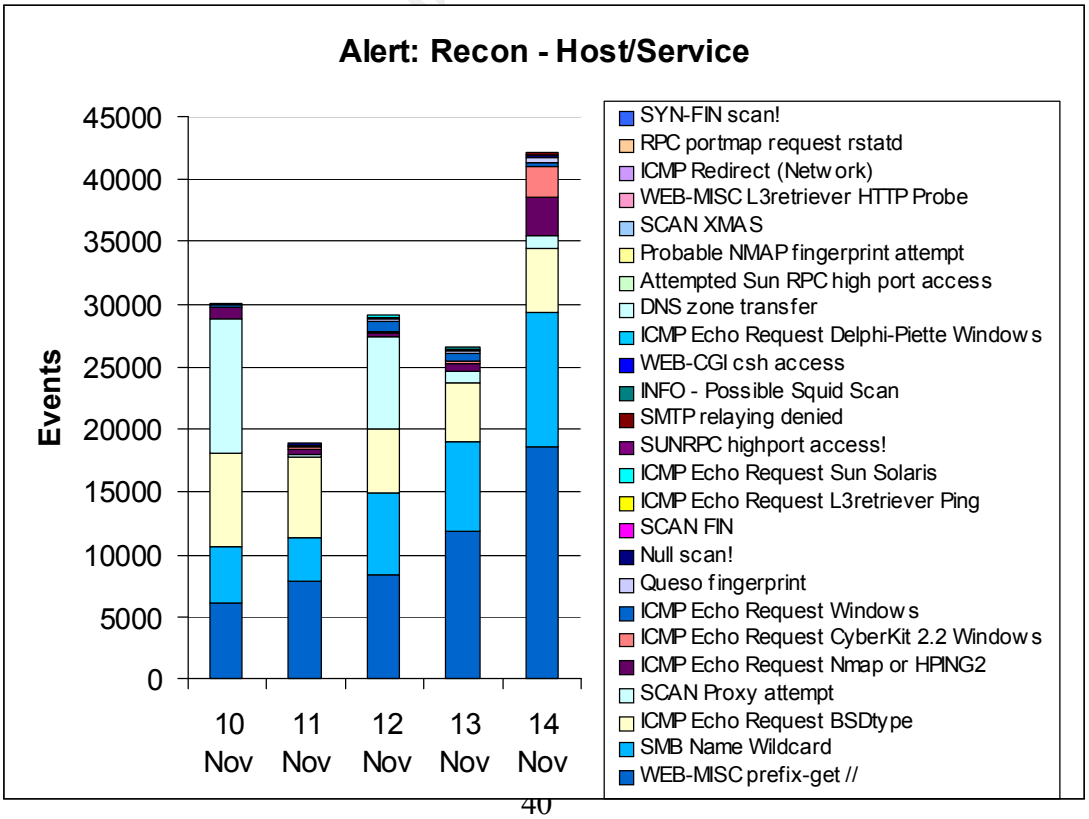
Both Graph 2 and Graph 3 show a steady increase in scanning activity.

Due to the large amount of scanning events attributed to the four hosts listed above, further analysis of these hosts is warranted (See "Analysis"). Half of the ten external hosts listed above originate from the same IP address block: 205.188.0.0/16. This netblock also requires further analysis (See "Analysis").

Several alert event types also fall into the Reconnaissance category. The graphs below show their breakdown.



Graphs 4,5: Alert Events: Reconnaissance



Alert: WEB-MISC prefix-get //
Threat: Information Gathering
Risk Level: Medium
Recommendation: Review MY.NET.253.114 web server logs for inappropriate access

SNORT Rule: (web-misc.rules)
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-MISC prefix-get //";flags: A+; uricontent:"get //"; nocase; classtype:attempted-recon; sid:1114; rev:2;)

This technique may allow an attacker to gather information about a target web server such as web server software and version. This could be used to focus a subsequent attack. Over 1,600 source IP addresses generated this event directed against the following hosts:

MY.NET.100.165	MY.NET.150.83	MY.NET.179.77
MY.NET.253.114	MY.NET.253.115	MY.NET.60.11

A majority of the 52,734 events were directed against MY.NET.253.114. This host was also subject to other reconnaissance and exploit attempts. Alert events include:

- Queso fingerprint
- High port 65535 tcp - possible Red Worm - traffic
- Port 55850 tcp - Possible myserver activity - ref. 010313-1
- Possible trojan server activity
- IDS475/web-iis_web-webdav-propfind
- spp_http_decode: CGI Null Byte attack detected
- spp_http_decode: IIS Unicode attack detected
- Watchlist 000222 NET-NCFC
- WEB-CGI archie access
- WEB-CGI redirect access
- WEB-CGI rsh access
- WEB-CGI scriptalias access
- WEB-FRONTPAGE _vti_rpc access
- WEB-IIS _vti_inf access
- WEB-MISC 403 Forbidden
- WEB-MISC http directory traversal
- WEB-MISC ICQ Webfront HTTP DOS

This web server was consistently targeted throughout the period being reviewed. More information is required to determine if this host has been compromised. If this host and the information it houses are considered mission critical, The University should consider implementing more rigorous review and maintenance procedures, such as timely review of web server logs, business continuity planning and implementation, and incident response.

Netblock: 205.188.0.0/16
Threat: Information gathering, network congestion
Risk Level: High
Recommendation: Place this netblock in a watchlist; block IP range at border router

Six of the top ten internal talkers originated from this netblock. Hosts 205.188.246.121, 205.188.233.185, 205.188.233.121 and 205.188.244.57 were involved in a large UDP portscan across the entire MY.NET network.

WHOIS Information:

```
America Online, Inc (NETBLK-AOL-DTC)
   22080 Pacific Blvd
   Sterling, VA 20166
   US

   Netname: AOL-DTC
   Netblock: 205.188.0.0 - 205.188.255.255

   Coordinator:
      America Online, Inc.  (AOL-NOC-ARIN)  domains@AOL.NET
      703-265-4670
```

The large number of alerts generated by these hosts might indicate a denial of service or forged packets. However, all scans were directed at specific, well-known ports such as 137 (netbios), 6970 (RealAudio), and 53 (DNS). The University should consider placing this netblock in a watchlist, or even blocking these IPs at the border router.

Host: MY.NET.5.75
Threat: Possible Denial of Service
Risk Level: Medium (Availability)
Recommendation: Remove host from network; perform forensic analysis

Typical scan events for this host:
Nov 12 01:17:45 MY.NET.5.75:67 -> MY.NET.218.198:68 UDP
Nov 12 01:17:46 MY.NET.5.75:67 -> MY.NET.240.190:68 UDP
Nov 12 01:17:48 MY.NET.5.75:67 -> MY.NET.228.94:68 UDP
Nov 12 01:17:48 MY.NET.5.75:67 -> MY.NET.223.170:68 UDP
Nov 12 01:17:49 MY.NET.5.75:67 -> MY.NET.220.158:68 UDP
Nov 12 01:17:49 MY.NET.5.75:67 -> MY.NET.233.78:68 UDP

Over 2000 different destination IPs are spread through MY.NET.217-243.0/24. UDP 67 is a well known port for BOOTP server and UDP 68 is a well known port for the BOOTP client. BOOTP is often used to find an IP address for a diskless client attempting to boot over a network (see RFC 951). The traffic above would be indicative of a BOOTP server responding to a BOOTP client request. However, no stimulus event, such as a broadcast to port 68, was captured in the event logs under consideration.

No information could be found for trojans or viruses which use this port as a DoS. However, the large number of destination IPs and the high rate at which the packets are being generated indicates this may be a denial of service attack - over 2 million events are generated by this host during this five day period. Alternatively, this host may be misconfigured and flooding the network with bogus traffic. Finally, this may be a reconnaissance tool, left in place by an attacker to scan for vulnerable hosts. This is also unlikely since the scan is using UDP, an "unreliable" protocol not guaranteed to return information to the sender.

No OOS events were generated for this host.

Host: MY.NET.5.76
Threat: Possible Denial of Service
Risk Level: Medium (Availability)
Recommendation: Remove host from network; perform forensic analysis

This host has generated similar events to MY.NET.5.75:
Nov 12 00:00:01 MY.NET.5.76:67 -> MY.NET.204.130:68 UDP
Nov 12 00:00:02 MY.NET.5.76:67 -> MY.NET.210.102:68 UDP
Nov 12 00:00:02 MY.NET.5.76:67 -> MY.NET.204.150:68 UDP
Nov 12 00:00:02 MY.NET.5.76:67 -> MY.NET.211.86:68 UDP
Nov 12 00:00:02 MY.NET.5.76:67 -> MY.NET.201.130:68 UDP
Nov 12 00:00:03 MY.NET.5.76:56681 -> MY.NET.200.191:23 SYN ******S*
Nov 12 00:00:04 MY.NET.5.76:67 -> MY.NET.201.170:68 UDP

Approximately 1200 destination IPs spread through MY.NET.200-212.0/24 receive this traffic. Determination of intent is similar to the preceding host. This host also generates another scan event – TCP 23 SYN.

Nov 12 00:01:37 MY.NET.5.76:56709 -> MY.NET.200.191:23 SYN ******S*
Nov 12 00:01:48 MY.NET.5.76:56715 -> MY.NET.200.191:23 SYN ******S*
Nov 12 00:01:52 MY.NET.5.76:56718 -> MY.NET.200.192:23 SYN ******S*
Nov 12 00:03:43 MY.NET.5.76:56750 -> MY.NET.200.191:23 SYN ******S*
Nov 12 00:04:01 MY.NET.5.76:56758 -> MY.NET.200.147:23 SYN ******S*
Nov 12 00:04:21 MY.NET.5.76:56768 -> MY.NET.200.191:23 SYN ******S*
Nov 12 00:05:02 MY.NET.5.76:56778 -> MY.NET.200.184:23 SYN ******S*

Approximately such 10,000 events were generated. The 133 hosts receiving this traffic are all in the MY.NET.200.2 – MY.NET.200.221 address range. This scan could inform an attacker if TCP port 23 (Telnet) is active, and might give away information about the underlying operating system. This scan may be trying to hide in the volume of UDP traffic being generated by the host, but does not explain the multiple SYN scans to this small number of destination hosts. This scan may also be in response to the UDP scan.

No OOS events were generated for this host.

Host: MY.NET.160.114
Threat: Inappropriate use of University resources
Risk Level: Medium
Recommendation: Investigate host for inappropriate software

Typical scan events for this host:
Nov 12 00:10:41 MY.NET.160.114:999 -> 65.11.85.27:1079 UDP
Nov 12 00:10:46 MY.NET.160.114:888 -> 209.103.193.40:27005 UDP
Nov 12 00:10:50 MY.NET.160.114:888 -> 209.103.193.40:27005 UDP
Nov 12 00:10:54 MY.NET.160.114:888 -> 209.103.193.40:27005 UDP

Nov 13 00:00:15 MY.NET.160.114:888 -> 65.42.128.206:1455 UDP
Nov 13 00:00:20 MY.NET.160.114:888 -> 4.33.6.138:3312 UDP
Nov 13 00:00:22 MY.NET.160.114:888 -> 24.130.208.205:1891 UDP
Nov 13 00:00:28 MY.NET.160.114:888 -> 63.193.147.214:4856 UDP
Nov 13 00:00:28 MY.NET.160.114:999 -> 63.193.147.214:4873 UDP

This is a UDP scan (18685 unique destination ports) using privileged source port 888 or 999 and is relatively slow – one event every few seconds. One interesting pattern is obvious:

Nov 11 19:49:39 MY.NET.160.114:**999** -> 4.42.56.40:**24714** UDP
Nov 11 19:49:39 MY.NET.160.114:**888** -> 4.42.56.40:**24715** UDP
Nov 13 01:34:56 MY.NET.160.114:**888** -> 131.212.95.67:**1742** UDP
Nov 13 01:34:57 MY.NET.160.114:**999** -> 131.212.95.67:**1743** UDP
Nov 14 00:00:12 MY.NET.160.114:**888** -> 198.82.86.233:**4140** UDP
Nov 14 00:00:12 MY.NET.160.114:**999** -> 198.82.86.233:**4146** UDP
Nov 14 00:00:29 MY.NET.160.114:**888** -> 4.63.35.14:**1766** UDP
Nov 14 00:00:29 MY.NET.160.114:**999** -> 4.63.35.14:**1767** UDP

In some cases, changing source ports will cause the source host to send another UDP packet to the same destination host, but with a slightly different destination port.

Finally, several alerts were generated which correlated to the scan events in question:

```
Nov 10 21:49:08 MY.NET.160.114:999 -> 204.155.149.59:27005 UDP
 11/10-21:49:11.650319  [**] MISC traceroute [**] 204.155.149.59:27005 ->
MY.NET.160.114:999
 Nov 10 21:49:12 MY.NET.160.114:999 -> 204.155.149.59:27005 UDP
 Nov 10 21:49:16 MY.NET.160.114:999 -> 204.155.149.59:27005 UDP
 11/10-21:49:16.706192  [**] MISC traceroute [**] 204.155.149.59:27005 ->
MY.NET.160.114:999
 11/10-21:49:17.065890  [**] MISC traceroute [**] 204.155.149.59:27005 ->
MY.NET.160.114:999
 11/10-21:49:17.155526  [**] MISC traceroute [**] 204.155.149.59:27005 ->
MY.NET.160.114:999
 11/10-21:49:17.596705  [**] MISC traceroute [**] 204.155.149.59:27005 ->
MY.NET.160.114:999
 Nov 10 21:49:20 MY.NET.160.114:999 -> 204.155.149.59:27005 UDP
```

Similar traffic (source port 27005, destination port 999 (or 888) was also noted between MY.NET.160.114 and 24.254.241.95. Port 27005 is associated with the first-person-shooter Half-Life.

No OOS events were generated for this host.

Host: 202.96.127.34
Threat: Medium
Risk Level: Medium
Recommendation: Create watchlist for 202.96.127.34
Investigate MY.NET.60.11 for possible compromise

Typical scan events generated by this host:
Nov 13 00:21:18 202.96.127.34:**4161** -> **MY.NET.60.11**:3896 UDP
Nov 13 00:21:18 202.96.127.34:**4161** -> **MY.NET.60.11**:14290 UDP
Nov 13 00:21:18 202.96.127.34:**4161** -> **MY.NET.60.11**:34081 UDP
Nov 13 00:21:18 202.96.127.34:**4161** -> **MY.NET.60.11**:37280 UDP
Nov 13 00:21:18 202.96.127.34:**4161** -> **MY.NET.60.11**:50363 UDP
Nov 13 00:21:18 202.96.127.34:**4161** -> **MY.NET.60.11**:20225 UDP
Nov 13 00:21:18 202.96.127.34:**4161** -> **MY.NET.60.11**:56015 UDP
Nov 13 00:21:18 202.96.127.34:**4161** -> **MY.NET.60.11**:16857 UDP

All 122,555 scan events had the same source port and destination IP address. 55,958 unique destination ports were recorded. All scan events were generated in a very short period of time, between 00:21:02 and 00:22:18 13 November.

Several alerts were also generated by this IP address. These can be correlated to scan events:

```
SCAN:Nov 13 00:21:06 202.96.127.34:4161 -> MY.NET.60.11:69 UDP
ALERT:11/13-00:21:06.973343  [**] TFTP - External UDP connection to internal
tftp server [**] 202.96.127.34:4161 -> MY.NET.60.11:69
SCAN:Nov 13 00:21:20 202.96.127.34:4161 -> MY.NET.60.11:69 UDP
ALERT:11/13-00:21:20.300434  [**] TFTP - External UDP connection to internal
tftp server [**] 202.96.127.34:4161 -> MY.NET.60.11:69
SCAN:Nov 13 00:21:51 202.96.127.34:4161 -> MY.NET.60.11:31337 UDP
ALERT:11/13-00:21:51.128749  [**] Back Orifice [**] 202.96.127.34:4161 ->
MY.NET.60.11:31337
ALERT:11/13-00:21:58.440287  [**] Port 55850 udp - Possible myserver activity
- ref. 010313-1 [**] 202.96.127.34:4161 -> MY.NET.60.11:55850
SCAN:Nov 13 00:22:01 202.96.127.34:4161 -> MY.NET.60.11:55850 UDP
ALERT:11/13-00:22:01.133514  [**] Port 55850 udp - Possible myserver activity
- ref. 010313-1 [**] 202.96.127.34:4161 -> MY.NET.60.11:55850
SCAN:Nov 13 00:22:05 202.96.127.34:4161 -> MY.NET.60.11:55850 UDP
ALERT:11/13-00:22:05.955018  [**] Port 55850 udp - Possible myserver activity
- ref. 010313-1 [**] 202.96.127.34:4161 -> MY.NET.60.11:55850
```

These alerts were generated due to the scan events (note the correlating time and source/destination address/port), and can most likely be considered "false positives."

No OOS events were generated by this host.

Secondary Host: MY.NET.60.11
Threat: High
Risk Level: High (Possible compromise)
Recommendation: Remove from network; perform forensic analysis

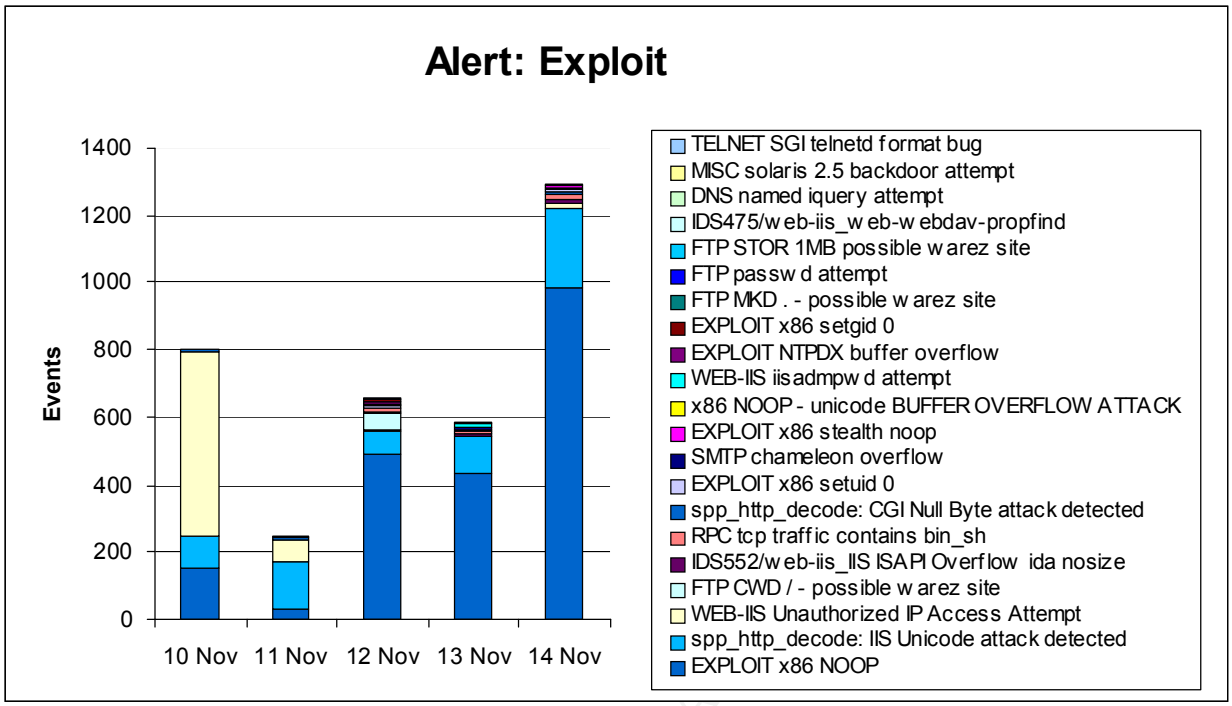
This host is being examined as a result of the analysis performed for MY.NET.5.75. Alert events show that this host both sends and receives extremely suspect traffic.

```
11/12-21:07:34.279800  [**] INFO Possible IRC Access [**] MY.NET.60.11:7065 -
> 64.214.30.92:6667
11/12-21:19:49.283039  [**] FTP MKD . - possible warez site [**]
24.249.225.207:2873 -> MY.NET.60.11:21
```

```
11/12-21:26:30.908520  [**] FTP MKD . - possible warez site [**]
24.249.225.207:2952 -> MY.NET.60.11:21
11/12-21:41:59.456440  [**] TELNET login incorrect [**] MY.NET.60.11:23 ->
24.252.67.84:63137
11/13-00:21:06.973343  [**] TFTP - External UDP connection to internal tftp
server [**] 202.96.127.34:4161 -> MY.NET.60.11:69
11/13-00:21:20.300434  [**] TFTP - External UDP connection to internal tftp
server [**] 202.96.127.34:4161 -> MY.NET.60.11:69
11/13-00:21:51.128749  [**] Back Orifice [**] 202.96.127.34:4161 ->
MY.NET.60.11:31337
11/13-00:21:58.440287  [**] Port 55850 udp - Possible myserver activity -
ref. 010313-1 [**] 202.96.127.34:4161 -> MY.NET.60.11:55850
11/13-00:22:01.133514  [**] Port 55850 udp - Possible myserver activity -
ref. 010313-1 [**] 202.96.127.34:4161 -> MY.NET.60.11:55850
11/13-00:22:05.955018  [**] Port 55850 udp - Possible myserver activity -
ref. 010313-1 [**] 202.96.127.34:4161 -> MY.NET.60.11:55850
11/13-00:54:42.788565  [**] TELNET login incorrect [**] MY.NET.60.11:23 ->
24.18.175.104:2933
11/13-01:35:37.266885  [**] TELNET login incorrect [**] MY.NET.60.11:23 ->
66.44.0.212:1053
11/13-01:54:39.878199  [**] INFO Possible IRC Access [**] MY.NET.60.11:8073 -
> 204.91.240.100:6667
11/13-04:03:00.371091  [**] ICMP Destination Unreachable (Fragmentation
Needed and DF bit was set) [**] 151.196.4.74 -> MY.NET.60.11
```

Exploitation

This category includes alerts which may indicate an exploit has been attempted. Alert types included in this category are shown in the graph below.



Graph 6: Alert: Exploit

Alert: x86 Buffer overflow attempts
Threat: Possible system compromise
Risk Level: Low, most likely false positive
Recommendation: None.

This class of alerts documents attempted buffer overflow exploits against Intel x86-based architecture. The alerts trigger on long strings of the Intel x86 "No operation" instruction, hex 0x90. In some cases it possible for these strings to show up in legitimate traffic, such as in graphics files.

Typical alert:
11/13-07:27:08.523953 [**] EXPLOIT x86 NOOP [**] 129.128.5.191:20 -> MY.NET.70.148:1819

The two hosts in the trace above account for a majority of these alert types. Source port 20 corresponds to the port over which FTP exchanges data with a client. The traces also show that the destination port is slowly increasing for each new log entry. This would be consistent with an FTP exchange; each new file that is transferred opens a new port on the client. Therefore, a majority of these alerts may be generating events based on the files being transferred between the FTP server and client.

Alert: spp_http_decode: IIS Unicode attack detected
Threat: Remote system compromise
Risk Level: High (Several internal "top talkers" received this attack)
Recommendation: Examine hosts for possible compromise; ensure patches are applied.

Due to a bug in the way IIS decodes URL strings, it is possible to traverse the web server host platform file system, run commands, etc. Please see http://www.wiretrip.net/rfp/p/doc.asp/i2/d57.htm or http://www.sans.org/infosecFAQ/threats/traversal.htm for a description.

Internal hosts receiving this attack:

MY.NET.100.165	MY.NET.253.114	MY.NET.5.25	MY.NET.5.75
MY.NET.10.253	MY.NET.253.115	MY.NET.5.33	MY.NET.5.76
MY.NET.110.92	MY.NET.253.118	MY.NET.5.4	MY.NET.5.88
MY.NET.111.140	MY.NET.253.119	MY.NET.5.43	MY.NET.5.92
MY.NET.112.246	MY.NET.253.123	MY.NET.5.44	MY.NET.5.95
MY.NET.11.4	MY.NET.253.125	MY.NET.5.45	MY.NET.60.14
MY.NET.1.2	MY.NET.253.127	MY.NET.5.46	MY.NET.60.22
MY.NET.140.2	MY.NET.253.23	MY.NET.5.59	MY.NET.6.14
MY.NET.179.77	MY.NET.253.24	MY.NET.5.64	MY.NET.6.16
MY.NET.253.106	MY.NET.5.121	MY.NET.5.66	MY.NET.6.7
MY.NET.253.109	MY.NET.5.19	MY.NET.5.67	MY.NET.70.186
MY.NET.253.112	MY.NET.5.248	MY.NET.5.74	

All servers should be reviewed to ensure the appropriate patches are applied (http://www.microsoft.com/technet/security/bulletin/fq00-057.asp) and all hosts should be examined for any possible compromise.

Several external hosts were also attacked by internal hosts:

Source	Destination	
MY.NET.98.153	194.67.23.248 194.67.18.5	This host seems to be using many different types of chat programs (MSN) and file sharing utilities (Gnutella, Kazaa). This activity correlates with http://www.incidents.org/archives/intrusions/msg01929.html. Recommend examination for malicious software.
MY.NET.153.127	211.229.209.115 211.229.209.80 211.229.209.87	This host performed a port 8080 SYN scan. Recommend examination for malicious software.
MY.NET.98.110	211.233.46.15	This host is also using Kazaa, and scan events show communication over 6112/UDP, attributed to the multiplayer game "Diablo. Recommend examination for malicious software.
MY.NET.98.190	217.170.71.33	Similar to events above. Recommend examination for malicious software.

Alert: FTP
Threat: Inappropriate use of University resources, possible system compromise
Risk Level: High
Recommendation: Ensure CWD and MKD are appropriate for listed FTP servers; examine servers for inappropriate content.

The "CWD" command allows the user to change the directory name prefix. MKD creates a directory in the ftp server. These commands are suspect, especially if followed by "/". This may be legitimate traffic. However, The University should investigate all FTP servers and determine if this level of access is appropriate. If not, these events should be considered hostile.

SNORT Rules: (policy.rules)
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP STOR 1MB possible warez site"; flags: A+; content:"STOR 1MB"; nocase; depth: 8; classtype:bad-unknown; sid:543; rev:1;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP RETR 1MB possible warez site"; flags: A+; content:"RETR 1MB"; nocase; depth: 8; classtype:bad-unknown; sid:544; rev:1;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP CWD / - possible warez site"; flags: A+; content:"CWD / "; nocase; depth: 6; classtype:bad-unknown; sid:545; rev:1;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP \"CWD \" possible warez site"; flags: A+; content:"CWD "; nocase; depth: 5; classtype:bad-unknown; sid:546; rev:1;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP \"MKD \" possible warez site"; flags: A+; content:"MKD "; nocase; depth: 5; classtype:bad-unknown; sid:547; rev:1;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP \"MKD . \" possible warez site"; flags: A+; content:"MKD ."; nocase; depth: 5; classtype:bad-unknown; sid:548; rev:1;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP \"MKD / \" possible warez site"; flags: A+; content:"MKD / "; nocase; depth: 6; classtype:bad-unknown; sid:554; rev:2;)

Host receiving MKD:

MY.NET.60.11	MY.NET.60.39	MY.NET.60.8

Hosts receiving CWD:

MY.NET.100.120	MY.NET.111.212	MY.NET.150.84	MY.NET.162.67
MY.NET.100.59	MY.NET.115.12	MY.NET.150.98	MY.NET.178.130
MY.NET.104.128	MY.NET.115.163	MY.NET.15.41	MY.NET.178.133
MY.NET.106.199	MY.NET.130.201	MY.NET.156.130	MY.NET.178.189
MY.NET.106.202	MY.NET.138.205	MY.NET.156.29	MY.NET.198.26
MY.NET.109.233	MY.NET.138.214	MY.NET.157.241	MY.NET.253.105
MY.NET.109.70	MY.NET.138.228	MY.NET.157.246	MY.NET.53.228
MY.NET.109.87	MY.NET.138.230	MY.NET.157.247	MY.NET.70.198
MY.NET.109.89	MY.NET.139.169	MY.NET.157.248	MY.NET.7.103
MY.NET.111.155	MY.NET.139.26	MY.NET.157.250	MY.NET.80.29
MY.NET.111.159	MY.NET.144.59	MY.NET.160.157	MY.NET.84.12
MY.NET.111.21	MY.NET.150.147	MY.NET.162.30	MY.NET.86.10

MY.NET.86.17	MY.NET.86.9	MY.NET.99.122
MY.NET.86.19	MY.NET.90.137	

The RETR verb asks the FTP server to send a specified file over an already established connection. In this case, two hosts attempted to retrieve the contents of the FTP server's password file. This is most likely a security violation. Investigation of server logs is required – a code "150" followed by "126" may indicate the file was successfully transferred, and may be compromised. The University should consider changing all passwords on the FTP server.

Snort Rule (ftp.rules):
alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"FTP passwd retreval attempt"; content:"RETR"; nocase; content:"passwd"; flags: A+; reference:arachnids,213; classtype:suspicious-filename-detect; sid:356; rev:2;)

Hosts receiving RETR passwd: MY.NET.253.105

Alert: IIS Unicode
Threat: Information leakage, possible system compromise
Risk Level: Medium
Recommendation: Ensure proper patches are installed on each host; examine web server logs for possible malicious activity

This alert is not generated by a specific SNORT rule, but by the HTTP preprocessor. HTTP requests which contain Unicode are first interpreted by the preprocessor. The result is sent back through the SNORT detection engine. Unicode my be used to traverse directories or run system commands via a web browser on some web servers. However, this alert may also be generated when interpreting legitimate multi-byte characters, such as Chinese.

Internal Hosts generating Unicode alerts:

MY.NET.100.165	MY.NET.253.114	MY.NET.5.25	MY.NET.5.75
MY.NET.10.253	MY.NET.253.115	MY.NET.5.33	MY.NET.5.76
MY.NET.110.92	MY.NET.253.118	MY.NET.5.4	MY.NET.5.88
MY.NET.111.140	MY.NET.253.119	MY.NET.5.43	MY.NET.5.92
MY.NET.112.246	MY.NET.253.123	MY.NET.5.44	MY.NET.5.95
MY.NET.11.4	MY.NET.253.125	MY.NET.5.45	MY.NET.60.14
MY.NET.1.2	MY.NET.253.127	MY.NET.5.46	MY.NET.60.22
MY.NET.140.2	MY.NET.253.23	MY.NET.5.59	MY.NET.6.14
MY.NET.179.77	MY.NET.253.24	MY.NET.5.64	MY.NET.6.16
MY.NET.253.106	MY.NET.5.121	MY.NET.5.66	MY.NET.6.7
MY.NET.253.109	MY.NET.5.19	MY.NET.5.67	MY.NET.70.186
MY.NET.253.112	MY.NET.5.248	MY.NET.5.74	

WHOIS queries of these external sites revealed they reside in either Russia or Korea. Both languages require Unicode support to view language characters properly. These alerts may be false positives.

External hosts generating Unicode alerts:

194.67.18.5	211.229.209.115	211.229.209.87	217.170.71.33
194.67.23.248	211.229.209.80	211.233.46.15	

Alert: WEB-IIS Unauthorized IP Access Attempt
Threat: Possible system compromise
Risk Level: Medium
Recommendation: Examine CS webserver logs for entries containing 195.92.168.167 or 195.92.168.163.

SNORT alert (web-iis.rules):
alert tcp $HTTP_SERVERS 80 -> $EXTERNAL_NET any (msg:"WEB-IIS Unauthorized IP Access Attempt"; flags: A+; content:"403"; content:"Forbidden\:"; classtype:web-application-attack; sid:1045; rev:2;)

This alert indicates that the web browser has asked for content which is forbidden. This could be a false positive if the client accidentally browsed a protected link. However, a majority of the 616 alerts were generated by one internal host (MY.NET.240.178) against two external hosts (195.92.168.167, 195.92.168.163). Unfortunately, no other alerts can be attributed to MY.NET.240.178. Both external hosts, however, generated three other alerts, including an internal watchlist alert:

```
11/11-18:20:34.021186  [**] CS WEBSERVER - external web traffic [**] 195.92.168.167:32164 ->
MY.NET.100.165:80
11/11-18:21:14.075743  [**] CS WEBSERVER - external web traffic [**] 195.92.168.167:37916 ->
MY.NET.100.165:80
11/11-19:58:35.041904  [**] WEB-FRONTPAGE _vti_rpc access [**] 195.92.168.167:54466 ->
MY.NET.179.77:80

11/14-19:05:06.499478  [**] CS WEBSERVER - external web traffic [**] 195.92.168.163:62392 ->
MY.NET.100.165:80
11/14-19:05:10.882129  [**] CS WEBSERVER - external web traffic [**] 195.92.168.163:62905 ->
MY.NET.100.165:80
11/14-19:05:20.199215  [**] CS WEBSERVER - external web traffic [**] 195.92.168.163:64007 ->
MY.NET.100.165:80
```

WHOIS information from RIPE:

```
inetnum:      195.92.168.0 - 195.92.171.255
netname:      E2-BRM-POP
descr:        Energis Squared Birmingham POP
descr:        In case of problems, please contact +44 113 2346068
descr:        Please do not send abuse reports to tech or admin contacts
descr:        Abuse reports to abuse@energis-squared.com please!
country:      GB
admin-c:      PJ3130-RIPE
tech-c:       PJ3130-RIPE
rev-srv:      earth.theplanet.net
rev-srv:      venus.theplanet.net
rev-srv:      pluto.theplanet.net
status:       ASSIGNED PA
notify:       ripe-adm@planet.net.uk
mnt-by:       AS5388-MNT
```

```
changed:      darrenh@energis-squared.com 20001123
source:       RIPE

route:        195.92.0.0/16
descr:        Planet Online Limited
descr:        The White House
descr:        Melbourne St.
descr:        Leeds LS2 7PS United Kingdom
origin:       AS5388
mnt-by:       AS5388-MNT
changed:      matthew@planet.net.uk 19960612
source:       RIPE

person:       Pedro Jones
address:      Energis Squared
address:      Melbourne St
address:      Leeds, LS2 7PS
phone:        +44 113 207 6000
fax-no:       +44 113 2345656
e-mail:       pedro.jones@energis-squared.com
nic-hdl:      PJ3130-RIPE
mnt-by:       AS5388-MNT
changed:      ripe-adm@planet.net.uk 20010920
source:       RIPE
```

If these IP addresses should not be accessing the CS webserver, they should be regarded as hostile. The University may consider sending an email to abuse@energis-squared.com to report the incidents. The University should also examine the CS webserver logs to determine the types of information requested by the external addresses in order to determine intent.

Alert: IDS552/web-iis_IIS ISAPI Overflow ida nosize
Threat: Possible system compromise
Risk Level: Medium
Recommendation: Ensure all IIS web servers are properly patched; examine systems for possible compromise.

From http://www.eeye.com/html/Research/Advisories/AD20010618.html: "

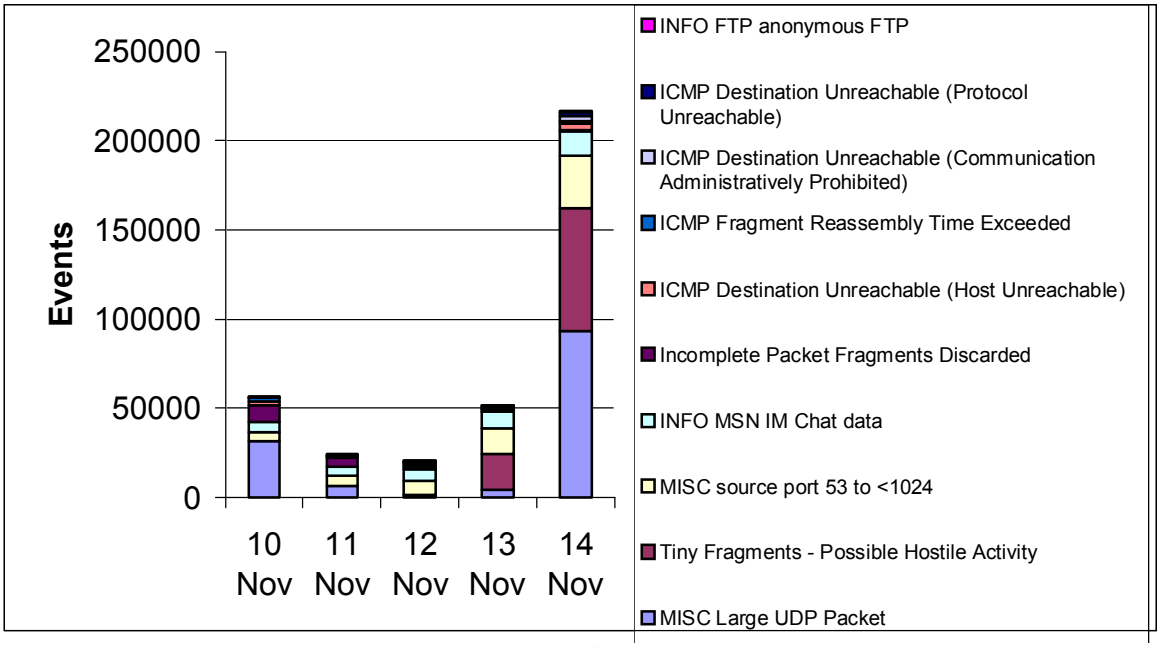
> The vulnerability lies within the code that allows a Web server to interact with Microsoft Indexing Service functionality. The vulnerable Indexing Service ISAPI filter is installed by default on all versions of IIS. The problem lies in the fact that the .ida (Indexing Service) ISAPI filter does not perform proper "bounds checking" on user inputted buffers and therefore is susceptible to a buffer overflow attack."

Hosts receiving IDS552/web-iis_IIS ISAPI Overflow ida nosize:

MY.NET.100.165	MY.NET.253.123	MY.NET.5.54	MY.NET.5.75
MY.NET.140.2	MY.NET.253.125	MY.NET.5.59	MY.NET.5.76
MY.NET.253.106	MY.NET.253.14	MY.NET.5.64	MY.NET.5.88
MY.NET.253.112	MY.NET.5.19	MY.NET.5.7	MY.NET.5.92
MY.NET.253.120	MY.NET.5.29	MY.NET.5.70	MY.NET.60.17

MY.NET.6.16 MY.NET.6.7 MY.NET.70.186

Two of the top talkers, MY.NET.5.75, and MY.NET.5.76, were targets of this attack.



Graph 7: Alerts: Suspect

Alert: MISC Large UDP Packet
Threat: Network congestion, denial of service
Risk Level: Medium
Recommendation: Place `61.150.0.0 - 61.150.31.255` into a watchlist

SNORT Rule (misc.rules):
alert udp $EXTERNAL_NET any -> $HOME_NET any (msg:"MISC Large UDP Packet"; dsize: >4000; reference:arachnids,247; classtype:bad-unknown; sid:521; rev:1;)

Typical Alert:
11/10-11:59:46.038679 [**] MISC Large UDP Packet [**] 61.150.5.19:2316 -> MY.NET.84.195:1293
11/10-11:59:46.144403 [**] MISC Large UDP Packet [**] 61.150.5.19:2316 -> MY.NET.84.195:1293
11/10-11:59:46.847479 [**] MISC Large UDP Packet [**] 61.150.5.19:2316 -> MY.NET.84.195:1293
11/10-11:59:46.920005 [**] MISC Large UDP Packet [**] 61.150.5.19:2316 -> MY.NET.84.195:1293
11/10-11:59:47.019914 [**] MISC Large UDP Packet [**] 61.150.5.19:2316 -> MY.NET.84.195:1293
11/10-11:59:48.147233 [**] MISC Large UDP Packet [**] 61.150.5.19:2316 -> MY.NET.84.195:1293

11/10-11:59:49.745045 [**] MISC Large UDP Packet [**] 61.150.5.19:2316 -> MY.NET.84.195:1293

A majority of these alerts were generated by 61.150.5.19 and most were directed against MY.NET.53.40. WHOIS information:

```
inetnum:     61.150.0.0 - 61.150.31.255
netname:     SNXIAN
descr:       xi'an data branch,XIAN CITY SHAANXI PROVINCE
country:     CN
admin-c:     WWN1-AP
tech-c:      WWN1-AP
mnt-by:      MAINT-CHINANET-SHAANXI
mnt-lower:   MAINT-CN-SNXIAN
changed:     ipadm@public.xa.sn.cn 20010309
source:      APNIC

person:      WANG WEI NA
address:     Xi Xin street 90# XIAN
country:     CN
phone:       +8629-724-1554
fax-no:      +8629-324-4305
e-mail:      xaipadm@public.xa.sn.cn
nic-hdl:     WWN1-AP
mnt-by:      MAINT-CN-SNXIAN
changed:     wwn@public.xa.sn.cn 20001127
source:      APNIC
```

This source host is also responsible for generating most of the "Incomplete Packet Fragments Discarded", and "ICMP Fragment Reassembly Time Exceeded" alerts. When these alert types were removed, several interesting log entries remained:

```
11/14-06:02:00.967110  [**] EXPLOIT x86 setuid 0 [**] 61.150.5.19:2392 ->
MY.NET.53.40:2660
11/14-07:09:37.654395  [**] EXPLOIT x86 setuid 0 [**] 61.150.5.19:2864 ->
MY.NET.53.40:2952
```

No alerts were generated by MY.NET.53.40 which indicate possible compromise. Without further network information, it is difficult to theorize why these alerts were generated. The large number of packets generated over an extended period of time indicate this may have been a denial of service, mainly targeting MY.NET.53.40. The traffic may also have been intended to hide the possible attack the two alerts above describe. The University should place this net block into a watchlist to track further activity. Blocking this IP address range at the border router or firewall may be prudent if the traffic continues.

Alert: Tiny Fragments
Threat: Denial of service
Risk Level: Medium
Recommendation: Examine MY.NET.8.1 for network problems; examine MY.NET.16.42 for possible compromise.

SNORT Rule (misc.rules):
alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"MISC Tiny Fragments"; fragbits:M; dsize: < 25; classtype:bad-unknown; sid:522; rev:1;)

Typical Alerts:
11/13-19:03:06.105050 [**] Tiny Fragments - Possible Hostile Activity [**] MY.NET.8.1 -> MY.NET.16.42
11/13-19:03:06.106681 [**] Tiny Fragments - Possible Hostile Activity [**] MY.NET.8.1 -> MY.NET.16.42
11/13-19:03:06.108205 [**] Tiny Fragments - Possible Hostile Activity [**] MY.NET.8.1 -> MY.NET.16.42

Nearly all alerts were identical to those shown above. No OOS alerts were generated by either of these hosts. No other alerts can be attributed to MY.NET.8.1. Here is an excerpt from the SNORT-users mail list from Martin Roesch (May 14,2000):

> "What have you got your minfrag preprocessor threshold set to? The minfrag option checks the size of IP fragments. If a fragment is smaller than a set threshold value, an alert is generated.
>
> The concept here is that no commercial network equipment that I've ever heard of fragments their traffic to less than 256 bytes, and so anything you see below that threshold value is probably *very* suspicious. FYI, nmap and fragrouter fragment to either 8 or 24 byte fragments. Judging by the volume of alerts you're seeing here, you're either under attack or something is broken."

Further analysis with TCPDump would be required to examine the actual packets. MY.NET.8.1 should be examined for problems.

MY.NET.16.42, however, is quite active and has generated alerts which indicate it may be compromised.

11/10-04:53:00.635736 [**] spp_http_decode: CGI Null Byte attack detected [**] MY.NET.16.42:24321 -> MY.NET.11.4:80
11/11-10:11:49.095231 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.16.42:42153 -> MY.NET.112.246:80
11/13-13:08:04.132957 [**] spp_portscan: portscan status from MY.NET.16.42: 20 connections across 20 hosts: TCP(20), UDP(0) [**]
11/13-13:08:05.710024 [**] spp_portscan: portscan status from MY.NET.16.42: 4 connections across 4 hosts: TCP(4), UDP(0) [**]

The University should remove MY.NET.16.42 from the network and examine it for possible compromise.

Alert: MISC source port 53 to <1024
Threat: Possible system compromise, information leakage
Risk Level: Medium, possible false positive
Recommendation: DNS servers should be considered mission critical; continued monitoring, perhaps creating an internal watchlist, is advised

SNORT Rule (misc.rules):
alert tcp $EXTERNAL_NET 53 -> $HOME_NET :1023 (msg:"MISC source port 53 to <1024"; flags:S; reference:arachnids,07; classtype:bad-unknown; sid:504; rev:2;)

Typical Alerts:
11/10-01:26:47.957278 [**] MISC source port 53 to <1024 [**] 204.189.73.2:53 -> MY.NET.1.3:53
11/10-01:26:52.569535 [**] MISC source port 53 to <1024 [**] 128.176.0.12:53 -> MY.NET.1.5:53
11/10-01:27:26.624393 [**] MISC source port 53 to <1024 [**] 64.124.237.67:53 -> MY.NET.1.4:53

Most of these alerts were generated with MY.NET.1.3, MY.NET.1.4, MY.NET.1.5 as destination IP addresses. Interaction between name servers occurs over port 53, and will cause this alert to false positive. The University should ensure that all destination hosts are name servers.

Destination hosts receiving this alert:

MY.NET.1.2	MY.NET.1.5	MY.NET.88.88	MY.NET.137.7
MY.NET.1.3	MY.NET.1.8	MY.NET.100.208	MY.NET.144.25
MY.NET.1.4	MY.NET.1.9	MY.NET.130.122	MY.NET.153.199

The internal host MY.NET.1.5 also received this exploit attempt:
11/10-18:30:00.325184 [**] DNS named iquery attempt [**] 210.95.176.193:2939 -> MY.NET.1.5:53

However no other alerts indicate that the host was compromised.

This host was the target of a large "SMB Name Wildcard" scan by MY.NET.219.30. An excellent description of this information gathering technique is http://www.sans.org/newlook/resources/IDFAQ/port_137.htm.

The external host 195.70.36.27 also attempted a zone transfer. Zone transfers are intended to keep primary and secondary name server information synchronized. However, misconfigured servers will allow zone transfers to any requesting host. This is often used by attackers to collect reconnaissance information before a network attack. Instructions on limiting zone transfers for Microsoft DNS servers can be found http://is-it-true.org/nt/atips/atips329.shtml. The host that performed the zone transfer has no other alert, scan, or OOS log entries. WHOIS information:

```
inetnum:      195.70.36.0 - 195.70.36.255
```

```
netname:        INTERWARE
descr:          InterWare Ltd.
descr:          IPs for Server Hosting
country:        HU
admin-c:        JA2447-RIPE
tech-c:         JA2447-RIPE
rev-srv:        ns1.interware.hu
rev-srv:        ns2.interware.hu
status:         ASSIGNED PA
mnt-by:         AS8358-MNT
changed:        angelo@interware.hu 20010507
source:         RIPE

route:          195.70.32.0/19
descr:          InterWare Ltd.
descr:          HU
origin:         AS8358
notify:         net-admin@interware.hu
mnt-by:         AS8358-MNT
changed:        angelo@interware.hu 20001115
source:         RIPE

person:         Janos Angeli
address:        InterWare Ltd.
address:        Victor Hugo u. 18-22.
address:        H-1132 Budapest
address:        Hungary
phone:          +36 1 3506892
fax-no:         +36 1 3506417
e-mail:         angelo@interware.hu
nic-hdl:        JA2447-RIPE
notify:         angelo@interware.hu
changed:        angelo@interware.hu 20000719
source:         RIPE
```

Alert: Peer-to-peer, chat, file sharing, gaming, IRC
Threat: Inappropriate use of University resources
Risk Level: Low - Medium
Recommendation: If this traffic is prohibited by University policy, block well known ports and continue to monitor alerts; if this traffic is allowed, remove rules from SNORT to reduce work load.

Slightly less than 21,000 alert log entries, and several thousand scan log entries were generated by various chat programs, IRC clients, Internet gaming protocols, and file sharing programs. If this traffic is allowed by University "appropriate use" policies, the University should consider removing these rules from the SNORT database, or monitoring for specific access paths, such as connections initiated from the Internet. If this traffic is not allowed, the University could block the well-known ports which are associated with these protocols, and continue to monitor for access attempts.

If the University does intend to continue monitoring these services, please be aware that several protocols are not captured by the SNORT rule set, including KaDzA

(http://www.incidents.org/archives/intrusions/msg01931.html), which communicates over 1214/tcp, and various Internet gaming protocols such as Diablo (6112/TCP). These tools will also generate false postive Scan log entries.

The several hundred IP addresses which alerted this activity will be attached if required.

Alert: Web server information gathering
Threat: Attackers may gain insight into web server software and file structure
Risk Level: Medium
Recommendation: Remove all default-installed functionality; disallow file system traversal; regularly review web server logs

Several techniques can be used by an attacker to retrieve information about a web server. The alert types considered here include:

- WEB-MISC 403 Forbidden
- WEB-MISC Attempt to execute cmd
- WEB-CGI scriptalias access
- WEB-MISC http directory traversal
- WEB-IIS _vti_inf access
- WEB-FRONTPAGE _vti_rpc access
- WEB-MISC count.cgi access
- WEB-CGI redirect access
- WEB-CGI rsh access
- WEB-MISC compaq nsight directory traversal
- WEB-MISC Lotus Domino directory traversal
- WEB-FRONTPAGE fpcount.exe access
- WEB-CGI formmail access
- WEB-FRONTPAGE shtml.exe
- WEB-CGI archie access
- WEB-CGI glimpse access
- WEB-CGI tsch access
- WEB-FRONTPAGE service.cnf access
- WEB-MISC guestbook.cgi access
- WEB-CGI w3-msql access
- WEB-CGI finger access
- WEB-CGI survey.cgi access
- WEB-CGI webgais access
- WEB-FRONTPAGE access.cnf access
- WEB-IIS .cnf access
- WEB-MISC Invalid URL

These alerts comprise approximately 4,000 events, and illustrate the types of information gathering techniques a potential attacker may use, and are indicative of a potential attacker accessing content which is often installed by default on Microsoft IIS web servers. It is unknown whether these events were false positive alerts generated by allowed traffic, thus their inclusion in this section. Listed below are all hosts which were recipients one or more WEB-type alerts, directed towards port 80/tcp, the well-known web server port:

MY.NET.100.165	MY.NET.253.118	MY.NET.5.19	MY.NET.5.46
MY.NET.140.2	MY.NET.253.119	MY.NET.5.248	MY.NET.5.59
MY.NET.150.83	MY.NET.253.123	MY.NET.5.25	MY.NET.5.64
MY.NET.179.77	MY.NET.253.125	MY.NET.5.29	MY.NET.5.66
MY.NET.253.106	MY.NET.253.127	MY.NET.5.33	MY.NET.5.67
MY.NET.253.109	MY.NET.253.23	MY.NET.5.4	MY.NET.5.74
MY.NET.253.112	MY.NET.253.24	MY.NET.5.43	MY.NET.5.75
MY.NET.253.114	MY.NET.5.121	MY.NET.5.44	MY.NET.5.76
MY.NET.253.115	MY.NET.5.13	MY.NET.5.45	MY.NET.5.88

MY.NET.5.92	MY.NET.60.17	MY.NET.6.16
MY.NET.5.95	MY.NET.60.22	MY.NET.6.7
MY.NET.60.14	MY.NET.6.14	MY.NET.70.186

In addition to analyzing the logs generated by this IDS, "best practice" would dictate that the University review and correlate logs for all internal web servers which generated an alert in the list above. Web server log entries have corresponding status codes (see RFC 2068 Section 6.1.1 at http://www.w3.org/Protocols/rfc2068/rfc2068 for a complete list). Status codes describe each HTTP request passing through a web server. At the very least, all 4XX and 5XX status codes should be investigated.

Attempting to perform this analysis by hand could be tiresome, especially for a busy server. Several tools to manage and analyze web server log entries are listed below:

- http://www.statslab.cam.ac.uk/~sret1/analog/: "**Analog** is a program to measure the usage on your web server. It tells you which pages are most popular, which countries people are visiting from, which sites they tried to follow broken links from, and all sorts of other useful information." Free.
- http://www.mrunix.net/webalizer: "***The Webalizer*** is a fast, free web server log file analysis program. It produces highly detailed, easily configurable usage reports in HTML format, for viewing with a standard web browser."
- http://www.webtrends.com: Commercial log analysis.

Properly securing a web server can be difficult. Here are some documents to assist web server administrators:

- http://www.sans.org/infosecFAQ/audit/IIS_sec.htm
- http://www.microsoft.com/technet/treeview/default.asp?url=/TechNet/prodtechnol/iis/tips/iis5chk.asp
- http://www.linuxplanet.com/linuxplanet/tutorials/1527/1/

Alert: Telnet
Threat: Inappropriate access to University resources
Risk Level: High
Recommendation: Confirm legitimacy of login attempts; examine system logs for malicious activity

Telnet is a protocol used to access a host remotely via the command line. Its use is discouraged as all session network traffic, including authentication information, is passed in the clear.

All events indicated internal hosts were accessing external hosts. Two alerts are considered.

Telnet login incorrect:
SNORT Rule (telnet.rules): alert tcp $EXTERNAL_NET any <- $HOME_NET 23 (msg:"TELNET login incorrect"; content:"Login incorrect"; flags: A+; reference:arachnids,127; classtype:bad-unknown; sid:718; rev:2;)

Typical traffic:
11/10-01:14:08.742040 [**] TELNET login incorrect [**] MY.NET.60.8:23 -> 131.118.250.172:4175
11/10-01:30:25.878389 [**] TELNET login incorrect [**] MY.NET.60.11:23 -> 64.7.51.94:10143
11/10-09:05:54.330302 [**] TELNET login incorrect [**] MY.NET.6.7:23 -> 24.18.175.188:2993

This may be allowed traffic. However, several hosts (listed below) attempted access to many different hosts. WHOIS information shows that the destination hosts are not related to The University. This may indicate malicious intent. The University must determine if this traffic is legitimate.

Telnet Source IP Addresses:

MY.NET.6.7	MY.NET.60.11	MY.NET.60.17	MY.NET.60.39
MY.NET.60.8	MY.NET.60.16	MY.NET.60.38	MY.NET.145.74

MY.NET.60.8, MY.NET.60.11, and MY.NET.60.39 are also noted in the previous FTP (Exploit) write-up.

MY.NET.60.17 was also accessed by a host from Watchlist 000220 on port 80.

If external Telnet access is not required, its use can be blocked by the firewall or border router.

Telnet Access:
Snort Rule (telnet.rules): alert tcp $EXTERNAL_NET any <- $HOME_NET 23 (msg:"TELNET access";flags: A+; content:"|FF FD 18 FF FD 1F FF FD 23 FF FD 27 FF FD 24|"; reference:arachnids,08; reference:cve,CAN-1999-0619; classtype:not-suspicious; sid:716; rev:1;)

The University should confirm that the Telnet access below is legitimate for these hosts:
MY.NET.60.40 -> 65.9.244.254
MY.NET.60.40 -> 65.14.236.126
MY.NET.6.46 -> 24.0.92.225

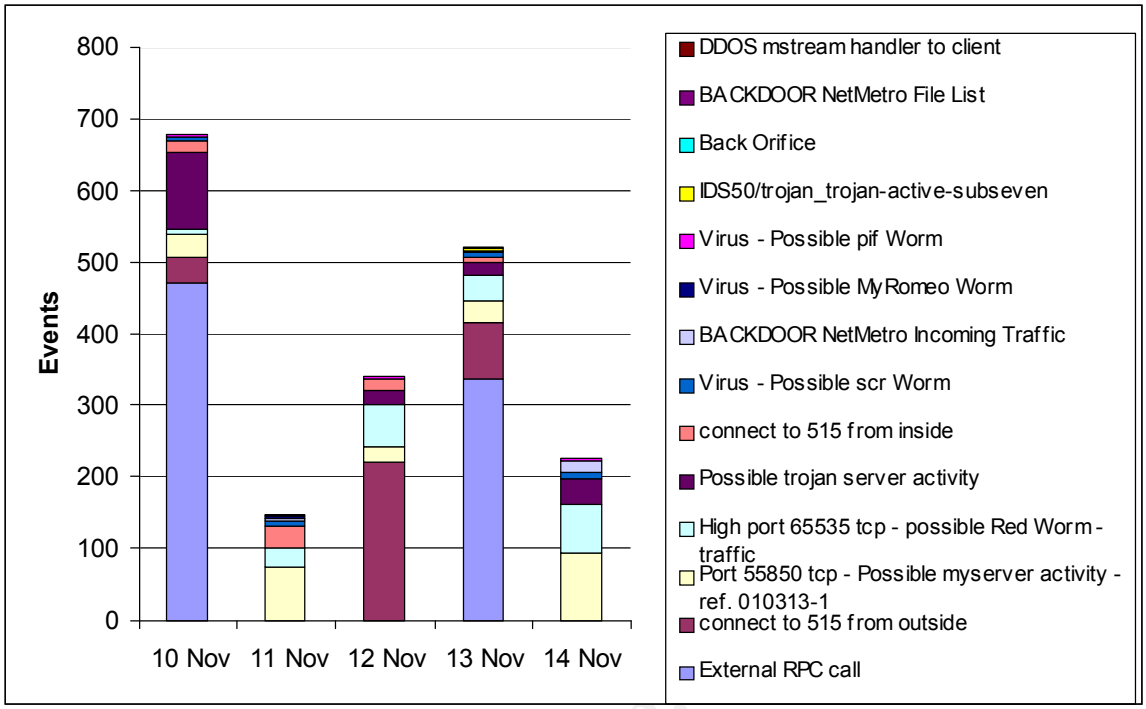
All destination hosts fall within the @Home cable network address range.

Alert: TFTP External access to Internal host
Threat: Disclosure of information
Risk Level: Medium
Recommendation: Block TFTP at Internet firewall; examine server system logs for file transfer

TFTP is a service which allows unauthenticated file transfer between hosts. Often used to boot diskless network clients, misconfigured TFTP servers may also allow clients to download any file within the host filesystem. The University should examine the following transactions for appropriateness:

Source Host -> Destination Host	WHOIS Information
63.150.23.120 -> MY.NET.158.102	RIVENET (NETBLK-QWEST-63-150-23-96) QWEST-63-150-23-96 63.150.23.96 - 63.150.23.127
200.176.87.59 -> MY.NET.130.45	Comite Gestor da Internet no Brasil (NETBLK-BRAZIL-BLK2) R. Pio XI, 1500 Sao Paulo, SP 05468-901 BR Netname: BRAZIL-BLK2 Netblock: 200.128.0.0 - 200.255.255.255 Maintainer: BR Coordinator: Registro.br (NF-ORG-ARIN) blkadm@nic.br +55 19 9119-0304
202.96.127.34 -> MY.NET.60.11	inetnum: 202.96.96.0 - 202.96.127.255 netname: CHINANET-ZJ descr: CHINANET Zhejiang province network descr: Data Communication Division descr: China Telecom country: CN admin-c: CH93-AP

If TFTP is not required, The University should block its use at the firewall.



Graph 8: Indications of Possible System Compromise (Without "High port 65535" alerts)

Alert: High port 65535 and 55850
Threat: Inappropriate use of University resources
Risk Level: Medium
Recommendation: Investigate destination host for online gaming software

Typical Alert:
11/11-00:05:40.673610 [**] High port 65535 udp - possible Red Worm - traffic [**]
66.79.17.223:65535 -> MY.NET.98.178:6112
11/11-00:05:40.673951 [**] High port 65535 udp - possible Red Worm - traffic [**]
66.79.17.223:65535 -> MY.NET.98.178:6112
11/11-00:05:41.100195 [**] High port 65535 udp - possible Red Worm - traffic [**]
MY.NET.98.178:6112 -> 66.79.17.223:65535

After infecting a host, the Red Worm would open a backdoor listening on port 65535. The rules relating to this alert generate an event based on any traffic sourced or destined for this port. In many cases, especially on busy servers, legitimate traffic may flow through this port, but will still generate a SNORT event. The host and port combinations shown above make up over 95% of these alert types. No malware correlated to these two ports. Port 6112/udp is often associated with the online game "Diablo." The alerts above were most likely generated by a Diablo client and server.

This should be investigated as an "appropriate use" issue. If University policies specifically prohibit this type of traffic, MY.NET.98.178 should be investigated and this port blocked at the firewall.

11/14-18:30:02.512988 [**] Port 55850 tcp - Possible myserver activity - ref. 010313-1 [**] MY.NET.6.47:55850 -> 209.132.220.133:25
11/14-18:27:17.901987 [**] Port 55850 tcp - Possible myserver activity - ref. 010313-1 [**] MY.NET.6.47:55850 -> 209.132.220.133:25

The majority of alerts are destined for a variety of well-known ports, such as mail and web services. In addition, no discernable correlation between source IPs could be discovered. These alerts appear to be false positives.

For reference, a list of well-known trojan ports can be found http://www.onctek.com/trojanports.html.

Alert: External RPC Call
Threat: Information leakage
Risk Level: Medium
Recommendation: Ensure system software is current; remove services which are not required.

Typical Alert:
11/13-04:30:16.471877 [**] External RPC call [**] 216.82.52.140:1946 -> MY.NET.190.248:111
11/13-06:05:56.009503 [**] External RPC call [**] 202.98.125.181:1400 -> MY.NET.132.1:111

This event often indicates RPC information gathering. Accessing the RPC portmapper will provide information about the types of RPC services available and which ports they listen. If RPC access is required to the Internet, access to the portmapper should be controlled through TCP wrappers. An excellent discussion on securing this service can be found http://nfs.sourceforge.net/nfs-howto/security.html#PORTMAPPER-SECURITY.

Four source hosts were responsible for generating all alert events. WHOIS information follows:

141.213.8.192

University of Michigan (NET-UMNET3)
Computer Aided Engineering Network (CAEN)
229 Chrysler Center
Ann Arbor, MI 48109-2092
US

Netname: UMNET3
Netblock: 141.213.0.0 - 141.213.255.255

Coordinator:
Killey, Paul M. (PMK5-ARIN)
paul@ENGIN.UMICH.EDU
(734) 763-4910 (FAX) (734) 936-3107

202.98.125.181

inetnum: 202.98.125.176 - 202.98.125.191
netname: TOPGROUP
descr: Sichuan TopGroup S&T Developping Co. Ltd
country: CN
admin-c: XS16-AP
tech-c: XS16-AP
mnt-by: MAINT-CHINANET-SC
changed: sxdong@mail.sc.cninfo.net 20000128
source: APNIC

216.194.26.87	MetTel, Inc. (NETBLK-METCONNECT-BLK-1) 44 Wall Street, 14th Floor New York, NY 10005 US Netname: METCONNECT-BLK-1 Netblock: 216.194.0.0 - 216.194.31.255 Maintainer: MTTL Coordinator: Metconnect (ZM116-ARIN) hostmaster@metconnect.net 212 607-2000
216.82.52.140	Internet Design Group (NETBLK-IDG-BLK) 1211 Semoran Blvd, Ste. 295 Casselberry, FL 32707 US Netname: IDG-BLK Netblock: 216.82.0.0 - 216.82.63.255 Maintainer: IDG Coordinator: Elliott, Mark (ME381-ARIN) ipadmin@durocom.com (859) 258-2537

These four hosts also generated significant scan events, specifically scanning for port 111/tcp focused on MY.NET.132.0/24. MY.NET.137.0/24, and MY.NET.190.0/24. There is a possibility only one of the hosts above was genuine, and the remaining were spoofed. No other events can be attributed to these hosts.

All hosts should be placed in watchlists to monitor for further activity.

Alert: Connect to 515 from outside
Threat: Information Leakage, possible system compromise
Risk Level: High
Recommendation: Ensure all suspect hosts are legitimate; place all suspect hosts on watchlists; remove service from host if not required; update system software; examine system for possible compromise

Typical Alert:
11/12-02:21:47.762603 [**] connect to 515 from outside [**] 144.132.185.106:4902 -> MY.NET.190.13:515
11/12-02:46:16.589616 [**] connect to 515 from outside [**] 255.255.255.255:31337 -> MY.NET.190.161:515
11/13-12:28:58.996982 [**] connect to 515 from outside [**] 211.220.193.241:3040 -> MY.NET.133.108:515

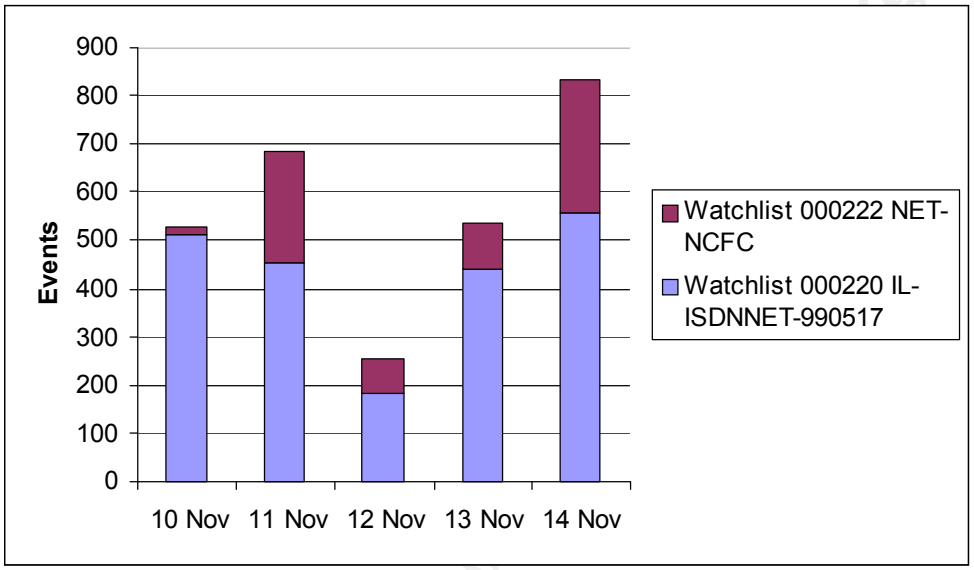
These alerts correlate to several port 515/tcp scans. A potential attacker will scan this port looking for vulnerable LPR services. An older reference can be found http://www.sans.org/newlook/alerts/port515.htm. No other events can be attributed to these hosts.

Three hosts are responsible for all alerts:

130.182.117.81	California State University, Los Angeles (NET-CSULANET) 5151 State University Drive Los Angeles, CA 90032 US Netname: CSULANET Netblock: 130.182.0.0 - 130.182.255.255 Coordinator: Gregorich, David T. (DTG11-ARIN) DTG@CSULA-PS.CALSTATELA.EDU (213) 343-2140
144.132.185.106	Telstra (NET-TELECOMAU2) 10/242 Exhibition st Melbourne, 3000 AU Netname: TELECOMAU2 Netblock: 144.132.0.0 - 144.132.255.255 Coordinator: TELSTRA CORPORATION (HM100-ORG-ARIN) hostmaster@broadway.bigpond.com +61 2 9395 9038
211.220.193.241	inetnum: 211.216.0.0 - 211.225.255.255 netname: KORNET descr: KOREA TELECOM

descr: KOREA TELECOM Internet Operating Center country: KR

External Watchlists



Graph 9: External Watchlists

Alert: Watchlist 000220 IL-ISDNNET-990517
Threat: Typically hostile traffic emanates from this address range
Risk Level: Medium
Recommendation: Continue to monitor activity; ensure destination hosts are fully patched and properly secured

Watchlists are used to flag any activity of particular interest to the IDS operator. In this case, traffic originating from 212.179.0.0/17 was flagged. All alerts are suspect. A breakdown of the most common alerts are detailed below:

Source	Destination	Activity
212.179.86.41	MY.NET.136, 137,138.0/24	Proxy scan, port 8080
212.179.127.29 212.179.81.223	MY.NET.150.145	Potential Kazaa traffic (port 1214/tcp)
212.179.85.100 212.179.84.55 212.179.81.47 212.179.43.110 212.179.85.79 212.179.82.25	MY.NET.150.133	Potential Kazaa traffic (port 1214/tcp)
212.179.87.229	MY.NET.150.220	Potential Kazaa traffic (port 1214/tcp)

212.179.81.220		
212.179.85.16	MY.NET.130.69	Potential Kazaa traffic (port 1214/tcp)
212.179.18.3	MY.NET.70.11	Potential Kazaa traffic (port 1214/tcp)
212.179.85.239	MY.NET.100.236	Potential Kazaa traffic (port 1214/tcp)
212.179.83.68 212.179.44.114 212.179.81.3	MY.NET.130.69	Potential Kazaa traffic (port 1214/tcp)
212.179.80.100	MY.NET.70.70	PsycWard Trojan well known port (Port 3777) http://www.dark-e.com/archive/trojans/psychward/big/index.shtml
212.179.43.110	MY.NET.150.133	Potential Kazaa traffic (port 1214/tcp)
212.179.81.134	MY.NET.115.115	Potential Kazaa traffic (port 1214/tcp)
212.179.43.96	MY.NET.104.76	Potential Kazaa traffic (port 1214/tcp)
212.179.85.159	MY.NET.60.17	HTTP

Most of these alerts appear to be Kazaa related.

Alert: Watchlist 000222 NET-NCFC
Threat: Watchlist traffic; inherently suspicious
Risk Level: Medium
Recommendation: Investigate network server logs for inappropriate access

The information below describes the types of interaction these watchlisted hosts had with internal hosts:

HTTP:
159.226.236.23:80 -> MY.NET.130.135:1236
159.226.250.54:80 -> MY.NET.99.220:35481
159.226.39.132:80 -> MY.NET.227.54:1570
159.226.42.11:80 -> MY.NET.97.246:1141
159.226.99.2:80 -> MY.NET.98.117:1146

FTP:
159.226.205.4:3354 -> MY.NET.136.111:23
159.226.45.204:3746 -> MY.NET.6.7:23

SMTP:
159.226.120.16:57032 -> MY.NET.253.42:25
159.226.21.3:32551 -> MY.NET.253.41:25
159.226.68.65:2548 -> MY.NET.253.42:25

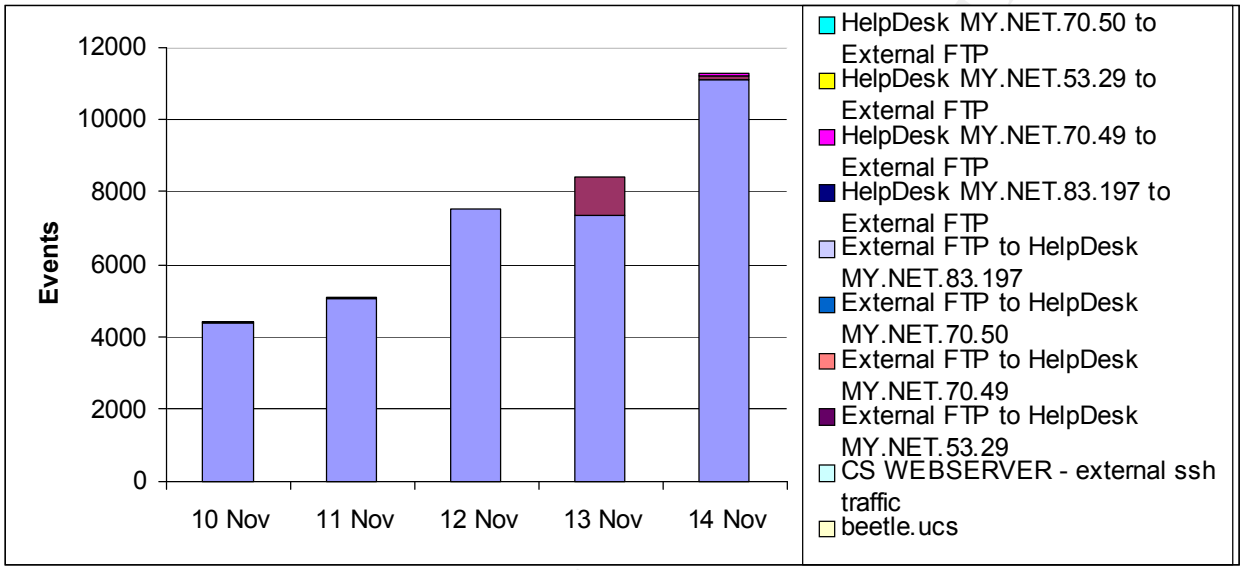
TELNET:
159.226.21.30:4382 -> MY.NET.100.165:21

AUTH:
159.226.120.16:113 -> MY.NET.253.42:42215
159.226.45.3:4203 -> MY.NET.6.7:113

159.226.5.222:1261 -> MY.NET.100.230:113
159.226.68.65:113 -> MY.NET.253.42:58188

Only a few specific services were accessed during this time period. However, the hostile nature of a watchlisted site is enough to warrant a closer look at each destination host.

Internal Watchlists



Graph 9: Internal Watchlists

Alert: CS Webserver
Three watchlists detailing ssh, ftp, and web access to the CS Webserver generated alert events.

Web (35,380 total events)
2,618 unique source IP addresses accessed this web server via the Internet. The top ten hosts are listed below:

Source Address	*Tally*
192.6.111.74	296
61.142.130.147	261
128.93.5.23	242
204.166.111.29	237
217.146.97.27	188
216.34.109.192	177
62.118.252.38	165
64.57.173.139	140
202.38.124.248	129
216.35.116.20	125

FTP (1263 total events)

200.61.154.158 generated the most events for this watchlist. 63 unique IP addresses accessed this FTP server. WHOIS information is listed below:

IFX Networks Argentina S.R.L. (NET-IFXNW-AR-1)
Av. Belgrano 1586 Piso 11
Capital Federal, Buenos Aires C 1093 AAQ
AR

Netname: IFXNW-AR-1
Netblock: 200.61.128.0 - 200.61.159.255
Maintainer: IFXA

Coordinator:
IFX Networks Argentina S.R.L. (ZI48-ARIN) operaciones.arin@ifxnw.com.ar
541146302400

Only the top ten hosts are listed below:

Source Address	***Tally***
200.61.154.158	998
213.7.160.2	36
202.54.26.125	27
217.4.4.94	16
213.140.14.135	16
216.209.174.123	14
136.205.103.103	14
129.89.253.137	14
194.167.168.1	11
130.236.132.241	9

SSH (3 total events)

Source Address	***Tally***
209.20.183.139	2
204.182.234.9	1

Alert: beetle.ucs

.The reason for the creation of this watchlist is unknown. However, all alerts contain MY.NET.70.69, the assumed address for beetle.ucs. Five different event types are observed (61 total events):

Possible Proxy Attempt (1):
11/10-00:16:37.949798 [**] beetle.ucs [**] 63.225.167.141:2611 -> MY.NET.70.69:8080

HTTP (43):

11/10-07:17:03.130699 [**] beetle.ucs [**] MY.NET.70.69:80 -> 200.225.204.222:3857
11/10-07:17:03.131780 [**] beetle.ucs [**] MY.NET.70.69:80 -> 200.225.204.222:3857
11/14-20:15:21.208838 [**] beetle.ucs [**] MY.NET.70.69:80 -> 130.203.168.121:2735

FTP (5):
11/10-22:25:12.114635 [**] beetle.ucs [**] MY.NET.70.69:21 -> 193.251.17.39:1714
11/12-18:40:16.641903 [**] beetle.ucs [**] 80.11.49.105:4034 -> MY.NET.70.69:21
11/12-18:15:00.705478 [**] beetle.ucs [**] 80.11.152.23:4124 -> MY.NET.70.69:21

SSH (9):
11/10-19:21:49.815789 [**] beetle.ucs [**] 195.27.130.2:22 -> MY.NET.70.69:22
11/11-01:22:43.888190 [**] beetle.ucs [**] MY.NET.70.69:22 -> 209.20.183.139:22
11/12-10:54:33.265658 [**] beetle.ucs [**] MY.NET.70.69:22 -> 206.142.53.21:22
11/13-10:48:57.250793 [**] beetle.ucs [**] 194.245.40.21:22 -> MY.NET.70.69:22
11/14-15:32:19.143911 [**] beetle.ucs [**] MY.NET.70.69:22 -> 204.182.234.9:3047

DNS (1):
11/11-03:52:11.947158 [**] beetle.ucs [**] MY.NET.70.69:53 -> 200.193.46.163:4041

Napster/IRC (2):
11/14-23:11:35.215745 [**] beetle.ucs [**] 206.167.75.78:6666 -> MY.NET.70.69:26821

Alert: External FTP <-> Help Desk
Four specific watchlists were created for this HelpDesk interaction. The reason for the creation of this watchlist is unknown. All alerts are shown:

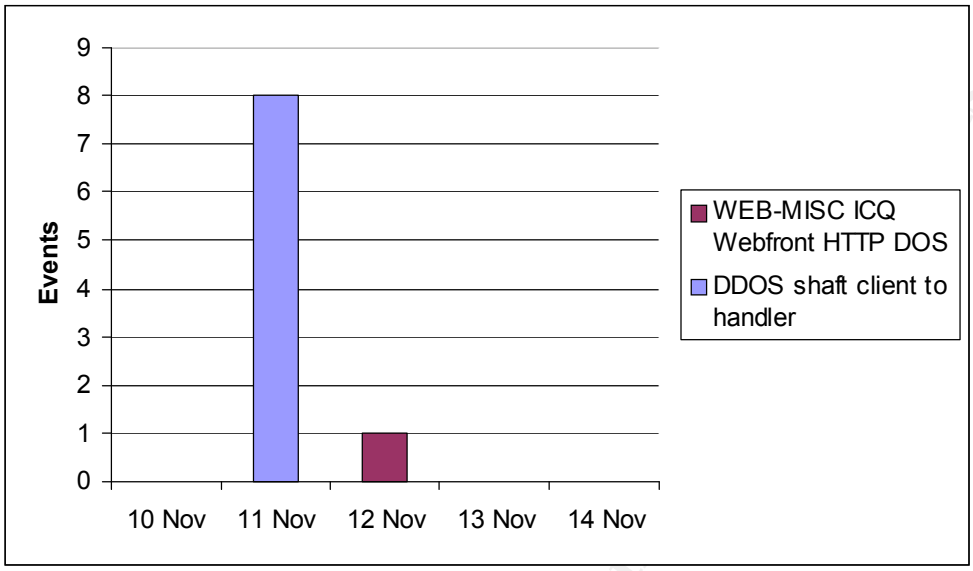
MY.NET.53.29
11/12-03:45:21.529486 [**] External FTP to HelpDesk MY.NET.53.29 [**]
212.68.218.130:4903 -> MY.NET.53.29:21
11/12-03:45:22.006782 [**] External FTP to HelpDesk MY.NET.53.29 [**]
212.68.218.130:4903 -> MY.NET.53.29:21
11/12-03:45:22.604220 [**] External FTP to HelpDesk MY.NET.53.29 [**]
212.68.218.130:4903 -> MY.NET.53.29:21
11/13-13:53:01.202876 [**] HelpDesk MY.NET.53.29 to External FTP [**]
MY.NET.53.29:4471 -> 161.69.2.7:21

MY.NET.70.49
11/10-01:00:37.449927 [**] External FTP to HelpDesk MY.NET.70.49 [**]
213.118.78.101:4293 -> MY.NET.70.49:21
11/11-11:32:03.183716 [**] External FTP to HelpDesk MY.NET.70.49 [**]
209.196.48.130:4580 -> MY.NET.70.49:21
11/12-18:15:03.545355 [**] External FTP to HelpDesk MY.NET.70.49 [**]
80.11.152.23:4104 -> MY.NET.70.49:21
11/14-09:10:28.775265 [**] HelpDesk MY.NET.70.49 to External FTP [**]
MY.NET.70.49:1041 -> 161.69.2.7:21

11/14-13:54:04.578864 [**] HelpDesk MY.NET.70.49 to External FTP [**]
MY.NET.70.49:2407 -> 199.199.2.81:21

MY.NET>70.50
11/12-18:15:00.608805 [**] External FTP to HelpDesk MY.NET.70.50 [**]
80.11.152.23:4105 -> MY.NET.70.50:21
11/14-03:24:11.859456 [**] External FTP to HelpDesk MY.NET.70.50 [**]
64.245.58.148:54103 -> MY.NET.70.50:21
11/14-05:25:18.064702 [**] External FTP to HelpDesk MY.NET.70.50 [**]
62.89.98.114:33161 -> MY.NET.70.50:21
11/14-08:37:04.270724 [**] HelpDesk MY.NET.70.50 to External FTP [**]
MY.NET.70.50:1038 -> 64.245.59.120:21

MY.NET.83.197
11/12-04:03:52.625065 [**] External FTP to HelpDesk MY.NET.83.197 [**]
212.68.218.130:4951 -> MY.NET.83.197:21
11/12-04:03:58.736993 [**] External FTP to HelpDesk MY.NET.83.197 [**]
212.68.218.130:4951 -> MY.NET.83.197:21
11/12-18:18:21.560211 [**] External FTP to HelpDesk MY.NET.83.197 [**]
80.11.152.23:3587 -> MY.NET.83.197:21
11/14-08:42:10.725912 [**] HelpDesk MY.NET.83.197 to External FTP [**]
MY.NET.83.197:1031 -> 161.69.2.7:21
11/14-08:52:35.884890 [**] HelpDesk MY.NET.83.197 to External FTP [**]
MY.NET.83.197:1093 -> 161.69.2.23:21
11/14-14:02:35.909082 [**] HelpDesk MY.NET.83.197 to External FTP [**]
MY.NET.83.197:1583 -> 161.69.2.23:21



Graph 10: Denial of Service

Alert: DDOS shaft client to handler
Threat: Internal hosts compromised by distributed denial of service zombies
Risk Level: Low
Recommendation: None

SNORT Rule (ddos.rules):
alert tcp $EXTERNAL_NET any -> $HOME_NET 20432 (msg:"DDOS shaft client to handler"; flags: A+; reference:arachnids,254; classtype:attempted-dos; sid:230; rev:1;)

Eight alerts were generated similar to this event:
11/11-06:58:47.244640 [**] DDOS shaft client to handler [**] 64.4.49.199:25 -> MY.NET.6.7:20432

No other alerts were generated by the source IP address. This is most likely a false positive; legitimate mail traffic was most likely flowing over port 20432.

No action is necessary.

Alert: WEB-MISC ICQ Webfront HTTP DOS
Threat: Crash HTTP server
Risk Level: Low
Recommendation: None

This one year-old denial of service targets a mini-HTTP server built into some ICQ clients. A reference can be found http://security-archive.merton.ox.ac.uk/bugtraq-200010/0119.html.

SNORT Rule (web-misc.rules):

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB-MISC ICQ Webfront HTTP DOS"; flags: A+; uricontent:"??????????"; classtype:web-application-attack; sid:1091; rev:3;)

One alert was generated for this rule:
11/12-05:28:38.715817 [**] WEB-MISC ICQ Webfront HTTP DOS [**] 196.41.219.40:3764 -> MY.NET.253.114:80

MY.NET.253.114 also received many hundreds of "WEB-MISC prefix-get //" events over this time period. Due to it's low occurrence, this alert may have been generated by non-malicious traffic. The source IP address generated no other events.

No action is required.

Out of Spec

The out of spec log collects events which have improper or impossible TCP/IP packet settings. For example, this packet dump shows that two reserved bits are "on," as well as SYN and FIN.

```
11/10-03:02:03.034869 24.170.11.227:6346 -> MY.NET.98.128:3517
TCP TTL:107 TOS:0x0 ID:55777  DF
21SF**** Seq: 0xA8B0687   Ack: 0x71CDCD   Win: 0x5018
18 CA 0D BD 0A 8B 06 87 00 71 CD CD 00 C3 50 18  .........q....P.
21 80 BB 63 00 00 18 41 03 56 9E 66 01 00 00 62  !..c...A.V.f...b
1C 16
```

This essentially tells the receiving host to setup and teardown the connection at the same time. It is anomalous events that will be described here.

Host: 199.183.24.194

This source host was responsible for nearly 70% of all OOS events. Most traffic was identical to the following packet dump:

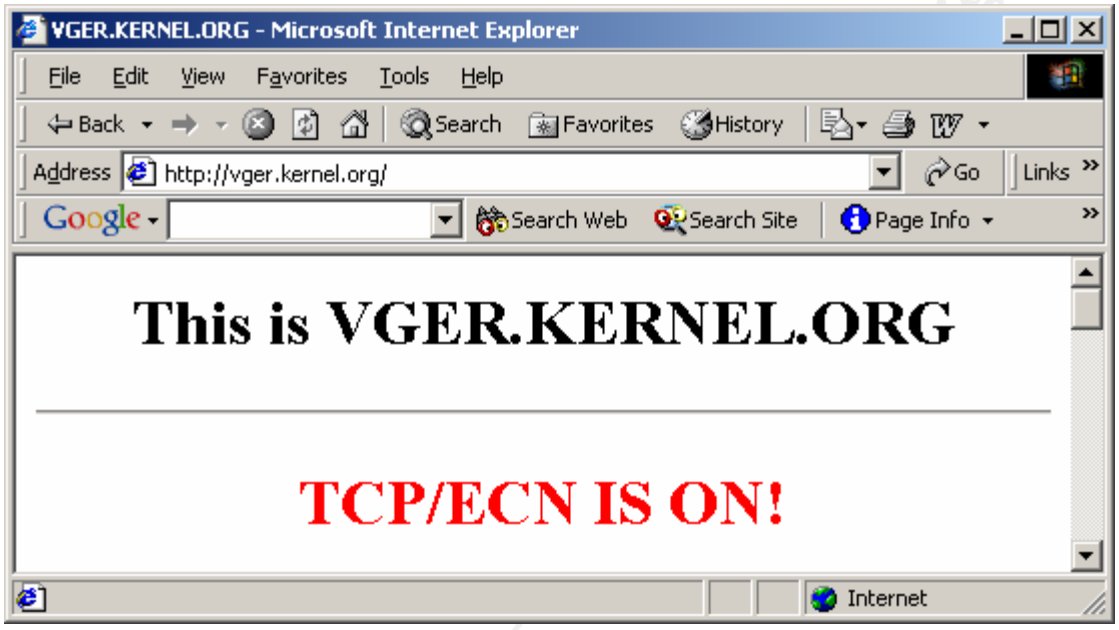
```
11/10-02:35:59.079619 199.183.24.194:53226 -> MY.NET.6.47:25
TCP TTL:52 TOS:0x0 ID:27227  DF
21S***** Seq: 0x9AFC41CD   Ack: 0x0   Win: 0x16D0
TCP Options => MSS: 1460 SackOK TS: 788343702 0 EOL EOL EOL EOL
```

These packets ended up in the OOS log because two reserved bits in the TCP header are "on." All alerts show these packets were headed for SMTP servers on port 25/tcp. Correlation with the alert logs enumerates the expected pattern:

```
11/10-14:08:22.426417  [**] Queso fingerprint [**] 199.183.24.194:50075 ->
MY.NET.6.47:25
11/10-14:20:22.195597  [**] spp_portscan: PORTSCAN DETECTED from
199.183.24.194 (STEALTH) [**]
11/10-14:20:24.090373  [**] spp_portscan: portscan status from
199.183.24.194: 1 connections across 1 hosts: TCP(1), UDP(0) STEALTH [**]
11/10-14:20:26.019424  [**] spp_portscan: End of portscan from
199.183.24.194: TOTAL time(0s) hosts(1) TCP(1) UDP(0) STEALTH [**]
```

This pattern repeats for every occurrence of an OOS log.

This source IP address resolves to vger.kernel.org. Upon further examination, it appears vger is using the Explicit Congestion Notification (ECN) protocol (see RFC 3168 at ftp://ftp.isi.edu/in-notes/rfc3168.txt).



Acording to the RFC:

```
This proposal specifies two new flags in the Reserved field of the TCP
header. The TCP mechanism for negotiating ECN-Capability uses the ECN-Echo
(ECE) flag in the TCP header. Bit 9 in the Reserved field of the TCP header
is designated as the ECN-Echo flag. The location of the 6-bit Reserved field
in the TCP header is shown in Figure 4 of RFC 793 [RFC793] (and is reproduced
below for completeness). This specification of the ECN Field leaves the
Reserved field as a 4-bit field using bits 4-7.

To enable the TCP receiver to determine when to stop setting the ECN-Echo
flag, we introduce a second new flag in the TCP header, the CWR flag. The CWR
flag is assigned to Bit 8 in the Reserved field of the TCP header.

  0   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|               |               | C | E | U | A | P | R | S | F |
| Header Length |   Reserved    | W | C | R | C | S | S | Y | I |
|               |               | R | E | G | K | H | T | N | N |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
Figure 4: The new definition of bytes 13 and 14 of the TCP Header.
```

According to http://www.sans.org/y2k/ecn.htm, “ECN uses the three – way handshake to determine whether or not a sender and receiver are ECN compatible. **During the initial SYN ECN will set TCP header bits 8 (CWR flag) and bit 9 (ECN –Echo flag)**, if the receiver of

this SYN is ECN compatible it will reply back in its SYN | ACK by setting TCP header bit 9. If the receiver is NOT compatible, the receiver will reply back by not setting any TCP header reserve bits. “

The packets captured here are ECN enabled. The University should consider updating their SNORT rule set with the following rule developed by Martin Roesch (http://archives.neohapsis.com/archives/snort/2001-01/0200.html):

alert tcp any any -> $HOME_NET any (tos: !0x02 flags: 12S; msg: "QUESO Fingerprint scan";)

Appendix A: Alert Event Tally Results

Alert: Recon/Network	10 Nov	11 Nov	12 Nov	13 Nov	14 Nov	Total
MISC traceroute	9801	8916	7951	11466	19968	**58102**
Spp portscan: end of portscan	2047	1780	1512	2140	3677	**11156**
ICMP traceroute	85	144	146	252	385	**1012**
NMAP TCP ping!	76	37	57	29	74	**273**
ICMP traceroute ipopts	0	0	1	0	0	**1**
Alert: Recon/Host	**10 Nov**	**11 Nov**	**12 Nov**	**13 Nov**	**14 Nov**	**Total**
WEB-MISC prefix-get //	6128	7852	8358	11774	18622	**52734**
SMB Name Wildcard	4443	3470	6488	7251	10673	**32325**
ICMP Echo Request BSDtype	7604	6434	5155	4622	5098	**28913**
SCAN Proxy attempt	10585	122	7397	989	1191	**20284**
ICMP Echo Request Nmap or HPING2	1023	541	384	737	3018	**5703**
ICMP Echo Request CyberKit 2.2 Windows	24	79	56	39	2365	**2563**
ICMP Echo Request Windows	124	126	856	695	307	**2108**
Queso fingerprint	87	76	109	177	504	**953**
Null scan!	72	181	84	104	144	**585**
SCAN FIN	2	12	116	6	7	**143**
ICMP Echo Request L3retriever Ping	9	4	8	59	57	**137**
ICMP Echo Request Sun Solaris	0	20	55	3	33	**111**
SUNRPC highport access!	23	12	14	44	13	**106**
SMTP relaying denied	2	5	13	2	32	**54**
INFO - Possible Squid Scan	8	7	6	7	21	**49**
WEB-CGI csh access	3	2	1	8	10	**24**
ICMP Echo Request Delphi-Piette Windows	1	14	3	2	2	**22**
DNS zone transfer	4	1	0	3	1	**9**
Attempted Sun RPC high port access	0	0	5	1	0	**6**
Probable NMAP fingerprint attempt	0	0	3	0	1	**4**
SCAN XMAS	0	0	2	0	0	**2**
WEB-MISC L3retriever HTTP Probe	0	0	0	0	2	**2**
ICMP Redirect (Network)	0	0	1	0	0	**1**
RPC portmap request rstatd	0	0	0	0	1	**1**
SYN-FIN scan!	0	0	1	0	0	**1**
Alert: Exploit	**10 Nov**	**11 Nov**	**12 Nov**	**13 Nov**	**14 Nov**	**Total**
EXPLOIT x86 NOOP	157	32	490	432	983	**2094**
spp_http_decode: IIS Unicode attack detected	95	140	70	111	239	**655**
WEB-IIS Unauthorized IP Access Attempt	541	63	1	1	10	**616**
FTP CWD / - possible warez site	0	0	52	0	0	**52**
IDS552/web-iis_IIS ISAPI Overflow ida nosize	1	3	6	9	12	**31**
RPC tcp traffic contains bin_sh	0	1	8	3	19	**31**
spp_http_decode: CGI Null Byte attack detected	4	5	4	2	5	**20**
EXPLOIT x86 setuid 0	2	1	3	5	8	**19**
SMTP chameleon overflow	1	1	3	4	6	**15**
EXPLOIT x86 stealth noop	1	1	3	3	3	**11**
x86 NOOP - unicode BUFFER OVERFLOW ATTACK	0	1	3	2	3	**9**

WEB-IIS iisadmpwd attempt	0	0	0	8	0	**8**
EXPLOIT NTPDX buffer overflow	0	0	6	0	1	**7**
EXPLOIT x86 setgid 0	0	0	2	3	2	**7**
FTP MKD . - possible warez site	0	0	4	1	0	**5**
FTP passwd attempt	0	0	2	0	0	**2**
FTP STOR 1MB possible warez site	0	0	0	0	2	**2**
IDS475/web-iis_web-webdav-propfind	0	0	1	0	1	**2**
DNS named iquery attempt	1	0	0	0	0	**1**
MISC solaris 2.5 backdoor attempt	0	0	0	1	0	**1**
TELNET SGI telnetd format bug	0	0	0	1	0	**1**
Alert: Suspect	**10 Nov**	**11 Nov**	**12 Nov**	**13 Nov**	**14 Nov**	**Total**
MISC Large UDP Packet	31360	6626	1593	4541	93053	**137173**
Tiny Fragments - Possible Hostile Activity	0	0	4	19642	69119	**88765**
MISC source port 53 to <1024	5438	5580	7610	14797	29314	**62739**
INFO MSN IM Chat data	5263	5171	6785	9114	13928	**40261**
Incomplete Packet Fragments Discarded	9732	5023	1506	232	909	**17402**
ICMP Destination Unreachable (Host Unreachable)	1791	429	921	1403	3089	**7633**
ICMP Fragment Reassembly Time Exceeded	2139	546	1086	115	1388	**5274**
ICMP Destination Unreachable (Communication Administratively Prohibited)	490	523	379	1006	2862	**5260**
ICMP Destination Unreachable (Protocol Unreachable)	188	185	214	812	2483	**3882**
INFO FTP anonymous FTP	467	252	598	194	351	**1862**
ICMP Destination Unreachable (Network Unreachable)	251	12	207	185	1175	**1830**
INFO Inbound GNUTella Connect accept	469	671	359	161	27	**1687**
WEB-MISC 403 Forbidden	158	175	285	369	456	**1443**
ICMP Source Quench	312	85	87	415	14	**913**
WEB-MISC Attempt to execute cmd	143	153	81	105	263	**745**
INFO Outbound GNUTella Connect accept	41	20	99	97	283	**540**
INFO Possible IRC Access	23	31	88	95	257	**494**
INFO Napster Client Data	17	17	29	54	278	**395**
WEB-CGI scriptalias access	93	6	82	180	8	**369**
WEB-MISC http directory traversal	40	53	51	65	150	**359**
WEB-IIS _vti_inf access	49	68	98	64	67	**346**
TELNET login incorrect	38	38	67	82	112	**337**
WEB-FRONTPAGE _vti_rpc access	44	74	92	56	68	**334**
FTP DoS ftpd globbing	0	61	67	161	0	**289**
WEB-MISC count.cgi access	13	21	47	65	111	**257**
INFO napster login	0	8	9	2	187	**206**
WEB-IIS view source via translate header	2	25	60	29	43	**159**
TCP SRC and DST outside network	8	19	31	33	61	**152**
INFO Inbound GNUTella Connect request	4	12	107	10	1	**134**
MISC Large ICMP Packet	17	5	26	41	28	**117**
TFTP - Internal TCP connection to external tftp server	0	21	29	6	59	**115**
WEB-CGI redirect access	5	6	10	29	24	**74**
Port 55850 udp - Possible myserver activity - ref. 010313-1	2	0	24	33	1	**60**
X11 outgoing	4	1	5	17	33	**60**

WEB-CGI rsh access	10	0	3	9	36	**58**
WEB-MISC compaq nsight directory traversal	13	7	4	10	6	**40**
ICMP Destination Unreachable (Fragmentation Needed and DF bit was set)	3	2	8	8	9	**30**
MISC PCAnywhere Startup	1	0	3	3	8	**15**
WEB-MISC Lotus Domino directory traversal	2	3	3	1	5	**14**
WEB-FRONTPAGE fpcount.exe access	2	4	0	5	2	**13**
RFB - Possible WinVNC - 010708-1	1	0	6	2	2	**11**
SNMP public access	0	3	0	2	5	**10**
Virus - Possible NAIL Worm	0	0	0	10	0	**10**
WEB-CGI formmail access	0	0	0	3	7	**10**
WEB-FRONTPAGE shtml.exe	0	1	1	4	4	**10**
TELNET access	0	1	3	1	2	**7**
WEB-CGI archie access	1	0	2	3	1	**7**
MISC Invalid PCAnywhere Login	0	0	0	0	6	**6**
TFTP - External TCP connection to internal tftp server	2	2	1	0	0	**5**
WEB-CGI glimpse access	3	2	0	0	0	**5**
WEB-CGI tsch access	0	1	0	1	3	**5**
ICMP Unassigned! (Type 7) (Undefined Code!)	0	0	0	3	0	**3**
WEB-FRONTPAGE service.cnf access	1	1	0	0	1	**3**
WEB-MISC guestbook.cgi access	3	0	0	0	0	**3**
TFTP - External UDP connection to internal tftp server	0	0	0	2	0	**2**
TFTP - Internal UDP connection to external tftp server	0	0	1	0	1	**2**
WEB-CGI w3-msql access	0	1	1	0	0	**2**
FTP .forward	0	0	0	0	1	**1**
ICMP IPV6 Where-Are-You	0	0	0	0	1	**1**
ICMP SRC and DST outside network	1	0	0	0	0	**1**
INFO napster upload request	0	0	0	0	1	**1**
INFO Outbound GNUTella Connect request	0	0	0	0	1	**1**
Virus - Possible shs Worm	0	0	0	0	1	**1**
Virus - SnowWhite Trojan Incoming	0	0	0	0	1	**1**
WEB-CGI finger access	0	0	1	0	0	**1**
WEB-CGI survey.cgi access	0	1	0	0	0	**1**
WEB-CGI webgais access	1	0	0	0	0	**1**
WEB-FRONTPAGE access.cnf access	0	0	1	0	0	**1**
WEB-IIS .cnf access	0	1	0	0	0	**1**
WEB-MISC Invalid URL	0	0	0	0	1	**1**
X11 xopen	0	0	1	0	0	**1**
Alert: Compromised Host	**10 Nov**	**11 Nov**	**12 Nov**	**13 Nov**	**14 Nov**	**Total**
High port 65535 udp - possible Red Worm - traffic	35	8054	30	66	131	**8316**
External RPC call	472	0	0	338	0	**810**
connect to 515 from outside	36	0	220	77	1	**334**
Port 55850 tcp - Possible myserver activity - ref. 010313-1	29	76	23	31	95	**254**
High port 65535 tcp - possible Red Worm -	9	25	59	37	66	**196**

traffic						
Possible trojan server activity	107	0	18	16	36	**177**
connect to 515 from inside	16	31	16	8	0	**71**
Virus - Possible scr Worm	5	7	1	6	8	**27**
BACKDOOR NetMetro Incoming Traffic	0	2	0	0	16	**18**
Virus - Possible MyRomeo Worm	1	4	1	3	0	**9**
Virus - Possible pif Worm	1	0	3	1	4	**9**
IDS50/trojan_trojan-active-subseven	1	2	0	1	0	**4**
Back Orifice	0	0	0	1	0	**1**
BACKDOOR NetMetro File List	0	0	0	1	0	**1**
DDOS mstream handler to client	0	0	0	1	0	**1**
Alert: Watchlist	**10 Nov**	**11 Nov**	**12 Nov**	**13 Nov**	**14 Nov**	**Total**
Watchlist 000220 IL-ISDNNET-990517	510	451	183	440	554	**2138**
Watchlist 000222 NET-NCFC	15	232	72	95	280	**694**
Alert: Internal Watchlist	**10 Nov**	**11 Nov**	**12 Nov**	**13 Nov**	**14 Nov**	**Total**
CS WEBSERVER - external web traffic	4369	5018	7496	7377	11120	**35380**
CS WEBSERVER - external ftp traffic	15	62	31	1051	104	**1263**
beetle.ucs	28	4	4	2	23	**61**
CS WEBSERVER - external ssh traffic	0	2	0	0	1	**3**
External FTP to HelpDesk MY.NET.53.29	0	0	3	0	0	**3**
External FTP to HelpDesk MY.NET.70.49	1	1	1	0	0	**3**
External FTP to HelpDesk MY.NET.70.50	0	0	1	0	2	**3**
External FTP to HelpDesk MY.NET.83.197	0	0	3	0	0	**3**
HelpDesk MY.NET.83.197 to External FTP	0	0	0	0	3	**3**
HelpDesk MY.NET.70.49 to External FTP	0	0	0	0	2	**2**
HelpDesk MY.NET.53.29 to External FTP	0	0	0	1	0	**1**
HelpDesk MY.NET.70.50 to External FTP	0	0	0	0	1	**1**
Alert: DOS	**10 Nov**	**11 Nov**	**12 Nov**	**13 Nov**	**14 Nov**	**Total**
DDOS shaft client to handler	0	8	0	0	0	**8**
WEB-MISC ICQ Webfront HTTP DOS	0	0	1	0	0	**1**

Appendix B: Event: Anonymous FTP server access

MY.NET.100.120	MY.NET.111.204	MY.NET.142.66	MY.NET.157.26
MY.NET.100.127	MY.NET.111.21	MY.NET.144.59	MY.NET.157.27
MY.NET.100.149	MY.NET.111.212	MY.NET.145.74	MY.NET.157.7
MY.NET.100.158	MY.NET.111.214	MY.NET.150.147	MY.NET.157.8
MY.NET.100.165	MY.NET.111.223	MY.NET.150.195	MY.NET.160.157
MY.NET.100.187	MY.NET.111.58	MY.NET.150.197	MY.NET.162.203
MY.NET.100.225	MY.NET.112.235	MY.NET.150.220	MY.NET.162.235
MY.NET.100.59	MY.NET.112.246	MY.NET.150.231	MY.NET.162.30
MY.NET.100.90	MY.NET.11.4	MY.NET.150.243	MY.NET.162.31
MY.NET.10.179	MY.NET.115.106	MY.NET.150.41	MY.NET.162.67
MY.NET.10.183	MY.NET.115.12	MY.NET.150.6	MY.NET.163.108
MY.NET.104.104	MY.NET.115.163	MY.NET.150.83	MY.NET.163.11
MY.NET.104.105	MY.NET.115.186	MY.NET.150.84	MY.NET.163.42
MY.NET.104.128	MY.NET.116.53	MY.NET.150.98	MY.NET.163.43
MY.NET.104.200	MY.NET.130.11	MY.NET.151.114	MY.NET.163.44
MY.NET.106.199	MY.NET.130.12	MY.NET.151.88	MY.NET.177.37
MY.NET.106.202	MY.NET.130.122	MY.NET.153.219	MY.NET.178.108
MY.NET.106.210	MY.NET.130.123	MY.NET.153.220	MY.NET.178.130
MY.NET.106.31	MY.NET.130.126	MY.NET.153.240	MY.NET.178.139
MY.NET.108.52	MY.NET.130.135	MY.NET.15.41	MY.NET.178.184
MY.NET.109.218	MY.NET.130.14	MY.NET.154.29	MY.NET.178.189
MY.NET.109.233	MY.NET.130.157	MY.NET.156.120	MY.NET.178.213
MY.NET.109.235	MY.NET.130.201	MY.NET.156.123	MY.NET.179.79
MY.NET.109.239	MY.NET.130.27	MY.NET.156.130	MY.NET.179.82
MY.NET.109.244	MY.NET.130.34	MY.NET.156.29	MY.NET.181.116
MY.NET.109.53	MY.NET.130.40	MY.NET.157.150	MY.NET.181.127
MY.NET.109.70	MY.NET.130.45	MY.NET.157.175	MY.NET.181.130
MY.NET.109.72	MY.NET.130.80	MY.NET.157.24	MY.NET.182.246
MY.NET.109.73	MY.NET.130.94	MY.NET.157.240	MY.NET.183.11
MY.NET.109.87	MY.NET.134.10	MY.NET.157.241	MY.NET.183.22
MY.NET.109.88	MY.NET.134.11	MY.NET.157.242	MY.NET.184.101
MY.NET.109.89	MY.NET.138.202	MY.NET.157.243	MY.NET.184.48
MY.NET.110.119	MY.NET.138.205	MY.NET.157.244	MY.NET.185.17
MY.NET.110.172	MY.NET.138.214	MY.NET.157.245	MY.NET.185.49
MY.NET.110.20	MY.NET.138.215	MY.NET.157.246	MY.NET.191.12
MY.NET.110.56	MY.NET.138.228	MY.NET.157.247	MY.NET.191.2
MY.NET.110.84	MY.NET.138.230	MY.NET.157.248	MY.NET.191.3
MY.NET.110.85	MY.NET.139.161	MY.NET.157.249	MY.NET.191.4
MY.NET.111.128	MY.NET.139.163	MY.NET.157.25	MY.NET.191.5
MY.NET.111.140	MY.NET.139.169	MY.NET.157.250	MY.NET.191.7
MY.NET.111.143	MY.NET.139.230	MY.NET.157.251	MY.NET.195.15
MY.NET.111.155	MY.NET.139.26	MY.NET.157.252	MY.NET.198.26
MY.NET.111.159	MY.NET.140.143	MY.NET.157.253	MY.NET.253.105
MY.NET.111.178	MY.NET.140.74	MY.NET.157.254	MY.NET.253.18

MY.NET.253.23	MY.NET.5.77	MY.NET.70.148	MY.NET.82.78
MY.NET.5.3	MY.NET.5.79	MY.NET.70.165	MY.NET.83.177
MY.NET.5.31	MY.NET.5.81	MY.NET.70.172	MY.NET.84.12
MY.NET.5.32	MY.NET.5.84	MY.NET.70.183	MY.NET.84.198
MY.NET.53.228	MY.NET.5.92	MY.NET.70.185	MY.NET.86.10
MY.NET.53.229	MY.NET.5.95	MY.NET.70.198	MY.NET.86.17
MY.NET.53.47	MY.NET.60.16	MY.NET.70.225	MY.NET.86.18
MY.NET.5.35	MY.NET.60.17	MY.NET.70.40	MY.NET.86.19
MY.NET.53.84	MY.NET.60.23	MY.NET.70.90	MY.NET.86.9
MY.NET.5.4	MY.NET.60.39	MY.NET.7.103	MY.NET.90.137
MY.NET.5.44	MY.NET.60.8	MY.NET.7.20	MY.NET.90.142
MY.NET.5.67	MY.NET.6.46	MY.NET.7.27	MY.NET.97.218
MY.NET.5.72	MY.NET.6.7	MY.NET.7.97	MY.NET.99.122
MY.NET.5.74	MY.NET.70.10	MY.NET.7.98	MY.NET.99.172
MY.NET.5.75	MY.NET.70.14	MY.NET.80.29	MY.NET.99.174

Appendix C: Data Reduction Methodology

The initial analysis of each log file was approached separately, since each offered different types of information:

- Scan logs – granular reconnaissance traces
- Alert logs – high level recon traces, detailed exploit/watchlist/etc/traces
- OOS – every host that was closely examined was checked for OOS file entries; stealth recon

"portscan" alerts were removed from consideration when building the final alert list since that data was represented to a greater degree of granularity in the scan logs.

In order to begin tallying, lists of unique IP addresses, ports, and alert event types had to be gathered. Like almost every other certification candidate, I made heavy use of grep, cut, and sort. Next, custom tools to tally various things were built. This Perl script tallies alert types and separates them by day:

```
#!/usr/bin/perl

$dir = "alert";

opendir(DIR, $dir) or die "Cannot open $dir: $!\n";
open(FILE, "alert_ip_list.txt") or die "Cannot open file: $!\n";

$i = 0;

while(<FILE>) {

        chomp;
        $scans[$i][0] = $_;
        $scans[$i][1] = 0;
        $scans[$i][2] = 0;
        $scans[$i][3] = 0;
        $scans[$i][4] = 0;
        $scans[$i][5] = 0;
        $scans[$i][6] = 0;

        $i++;

}

$file_count = 2;

while(defined($file = readdir(DIR))) {

        next if $file =~ /^\.\.?$/;

        print "Working $file...\n";

        open(SCANFILE, "$dir\/$file") or die "Cannot open $file: $!\n";

        while(<SCANFILE>) {

                /\-\>\s(\S+?)\:\S+?/;
                $scan_type = $1;
```

```
                for ($j=0; $j<@scans; $j++) {

                        if ($scans[$j][0] eq $scan_type) {

                                $scans[$j][1]++;
$scans[$j][$file_count]++;
                                next;

                        }

                }

        }

        close(SCANFILE);

        $file_count++;

}

open(OUTFILE, ">daily_alert_ip_outfile.txt") or die "Cannot open
daily_alert_ip_
outfile.txt: $!\n";

for ($k=0; $k<@scans; $k++) {

        print
"$scans[$k][0],$scans[$k][1],$scans[$k][2],$scans[$k][3],$scans[$k
][4],$scans[$k][5],$scans[$k][6]\n";
print OUTFILE "$scans[$k][0],$scans[$k][1],$scans[$k][2],$scans[$k][3],$
scans[$k][4],$scans[$k][5],$scans[$k][6]\n";

}

close(OUTFILE);
```

The tallies these tools produced provided a basis to begin analysis. The top ten internal and external talkers were generated from the scan and alert logs. Each alert type was categorized further into eight groups (see report) and the most alerted events or the most potential harmful events were analyzed.

Appendix D: References

Cox, Steve. "Portscan Websites." URL: http://kernel.org.in/mirrors/munitions/documents/linux-security-links.

"Testes de Seguranca." Seguranca Maxima. URL: http://br.geocities.com/segurancamaxima/smtest.htm.

"Security for Cable Networks." URL: http://www.dslreports.com/security/sec027.htm.

"wu-ftpd port contains remote root compromise [REVISED]." FreeBSD, Inc. 11 July 2000. URL: http://www.securityfocus.com/advisories/2374.

Shankar, Umesh, et al. "Detecting Format String Vulnerabilities with Type Qualifiers." University of California at Berkeley. 11 May 2001. URL: http://www.cs.berkeley.edu/~ushankar/research/percents/percents.pdf.

Computer Incident Response Capability, U.S. Department of Energy. "Information Bulletin." 26 June 2000. URL: http://ciac.llnl.gov/ciac/bulletins/k-054.shtml.

SecurityFocus. "wu-ftpd /bin SITE EXEC Misconfiguration Vulnerability." 30 Nov 1995. URL" http://www.securityfocus.com/bid/2241.

Vision, Max. URL: http://www.whitehats.com/info/IDS317.

Australian CERT. "wu-ftpd 'site exec' Vulnerability." 26 June 2000. URL: ftp://ftp.auscert.org.au/pub/auscert/advisory/AA-2000.02.

Nazario, Jose. "Detects Analyzed 6/28/00." Global Incidents Analysis Center. URL: http://www.incidents.org/archives/y2k/062800-1000.htm.

Johnson, John. 23 June 2000. URL: http://www.incidents.org/archives/y2k/062300-1430.htm.

Fox, Stuart. "Scan of the Month 19 Analysis." HoneyNet Project. October 2001. URL: http://project.honeynet.org/scans/scan19/scan/som28/analysis.html.

Unknown. URL: http://www.hot.ee/hagelberg/Hacks.htm.

Unknown. 31 Oct 2001. URL: http://www.ini2.net/mel/snort_trace/proxy-3110.html.

Incidents.org. "July 30, 2991 probes (part 2)." URL: http://www.incidents.org/archives/intrusions/msg01221.html.

Augustsson, John. 6 Feb 2001. URL: http://archives.neohapsis.com/archives/snort/2001-02/0073.html.

Foundstone, Inc. URL: http://www.foundstone.com/rdlabs/termsofuse.php?filename=superscan.exe.

Bevan Graham. URL: http://archives.neohapsis.com/archives/incidents/2001-06/0045.html.

Orebaugh, Angela. "Securing Solaris." SANS.org. 2 Oct 2000. URL: http://www.sans.org/infosecFAQ/unix/sec_solaris.htm.

O'Berry, Brian, Ritchey, Ronald W. "Securing Solaris from remote attack.: Nov 2000. Serverworld. URL: http://www.serverworldmagazine.com/sunserver/2000/11/attack.shtml.

Chounard, Jean. "How to install Solaris and have a good host security." 19 Nov 2000. URL: http://www.yassp.org.

Rainforestpuppy. "IIS %c1%1c bug." 28 Feb 2001. URL: http://www.wiretrip.net/rfp/p/doc.asp/i2/d57.htm.

Shields. Steven. "Web Server Folder Traversal vulnerability (MS00-076)." 13 Feb 2001. SANS Institute. URL: http://www.sans.org/infosecFAQ/threats/traversal.htm.

Microsoft Corp. "Microsoft Security Bulletin (MS00-057)." 2001. URL: http://www.microsoft.com/technet/security/bulletin/fq00-057.asp.

Unknown. "Greatly increased TCP port 1214 activity." Incidents.org. Oct 2001. URL: http://www.incidents.org/archives/intrusions/msg01929.html.

Hassell, Ruley, Permeh, Ryan. "All versions of Microsoft Internet Information Services Remote buffer overflow." EEye Digital Secuity. 18 Junr 2001. URL: http://www.eeye.com/html/Research/Advisories/AD20010618.html.

Alexander, Bryce. "Port 137 Scan." SANS Institute. 10 May 2000. URL: http://www.sans.org/newlook/resources/IDFAQ/port_137.htm.

"NT Admin Tip #329." URL: http://is-it-true.org/nt/atips/atips329.shtml.

URL: http://www.statslab.cam.ac.uk/~sret1/analog/
URL: http://www.mrunix.net/webalizer
URL: http://www.webtrends.com

Martinez, Patricia. "Internet Information Server Security Assessment: An Essential Audit." SANS Institute. 4 April 2001. URL: http://www.sans.org/infosecFAQ/audit/IIS_sec.htm.

Microsoft Corp. "Secure Internet Information Services 5 Checklist." 2001. URL: http://www.microsoft.com/technet/treeview/default.asp?url=/TechNet/prodtechnol/iis/tips/iis5chk.asp

Coar, Ken. "Security and Apache: An Essential Primer." LinuxPlanet.com. URL: http://www.linuxplanet.com/linuxplanet/tutorials/1527/1/

Boxmeyer, Jim. ONCTek. URL: http://www.onctek.com/trojanports.html.

Unknown. "Linux NFS-HOWTO." URL: http://nfs.sourceforge.net/nfs-howto/security.html#PORTMAPPER-SECURITY.

"Alert: Increased Probes toTCP port 515." 20 Nov 2000. URL: http://www.sans.org/newlook/alerts/port515.htm.

Chear, Charles. "ICQ WebFront HTTPd DoS." 7Oct 2000. URL: http://security-archive.merton.ox.ac.uk/bugtraq-200010/0119.html.

RFC 3168. URL: ftp://ftp.isi.edu/in-notes/rfc3168.txt.

Roesch, Martin. 12 Jan 2001. URL: http://archives.neohapsis.com/archives/snort/2001-01/0200.html.

Appendix E: TCP Destination Port Tally