Global Information Assurance Certification Paper

GIAC Intrusion Detection In Depth

Cisco IDS Network Appliance Installation

Network Detects and "Analyze This"

SANS September, 2002 New York
GCIA Practical Assignment v 3.3
Beth E. Binde
May 12, 2003

Abstract: This submission for the GCIA practical includes the following three sections:

- a case study on the installation of a Cisco Intrusion Detection System
- three network detects focusing on SOCKS proxy, DNS and FIN SCAN
- an extended analysis of 5 days of network logs (alerts, scans and "out of spec" packets) including security recommendations.

Assignment 1: Describe the State of Intrusion Detection
Cisco IDS Network Appliance Installation

Note that sources and references that would identify the target organization have been left out by design (not overlooked), and some details obfuscated as stipulated in the assignment directions.

Introduction

This case study is described in the context of the network topology of the target organization, as well as the rules and practices that govern the management of the network. Two Cisco Intrusion Detection System (IDS) sensor appliances (model IDS-4230-FE) were purchased during a period of active transition from a legacy network to a completely redesigned network topology. The results, both successes and failures, need to be understood in terms of the technical as well as the non-technical (or political) issues prevailing in the organization.

Constraints and Challenges

Some limited background as to the management and design of the network will enhance the understanding of the readers. At the start of the IDS project, the network was nearing completion of a redesign and upgrade to provide 100 MBPS to the desktop for faculty/staff and 10/100 MBPS to student residences. The design documentation from the project web site provides detailed information about the network implementation.

The basic design consists of isolated trees connected at their root nodes, utilizing four elements: core, distribution, access and leaf nodes (see Figure 1). The network relies on the Open Shortest Path First (OSPF) routing protocol.

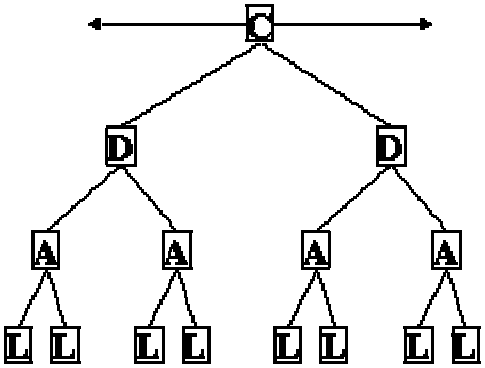


Figure 1: Topological map of the design network representing core (C), distribution (D), access (A), and leaf (L) nodes

- Core nodes are L3 devices representing the central network infrastructure.
- Core nodes are L3 neighbors to at least two other core nodes.
- Distribution nodes are L3 neighbors to at least one core node.
- Access nodes are L2 neighbors to at least one distribution node. Access nodes may not connect directly to core nodes.
- Leaf nodes are L2 neighbors to at least one access node. Leaf nodes may not connect directly to distribution nodes or core nodes.

In short, large or otherwise important buildings are designated as "A" buildings and have more direct access to the Internet handoff. However, the design guarantees that no site in the University will be more than 3 hops away from the gateway. It should be noted that the LAN infrastructure was moved to switches and VLANs (Virtual Local Area Networks) as part of the transition. Routers and switches are under centralized control rather than departmental control, although physically located in telephone closets in the various buildings.

A brief background on the terminology will be helpful to understanding the backbone network designed. L2 and L3 refer to the Open Systems Interconnect Reference Model (OSI-RM). Often referenced simply as "OSI", the model provides the basis for discussing the various elements of data communication. The lower three layers (layer 1, layer 2 and layer 3) address host-to-host communication functions. Layer 3, the network layer, is where actual delivery of data takes place. It provides delivery and addressing services, routing and forwarding. Layer 2, the data link, controls transport across the physical connection medium—the Media Access Control (MAC) and the link control (LC) or interface between Layer 3 protocols and the MAC. Layer 2 protocols include Ethernet and Fiber Distributed Data Interface (FDDI). [MICHAEL, chapter 1]

Given the immense size of the network (over 50,000 nodes) and the relatively small size of the staff, the network is subject to restrictions in an effort to ensure

robustness and consistency across applications. Departments are not permitted to connect firewalls (or any other devices) directly to routers. Neither may they be deployed between switches and routers. All traffic from the department must flow through a centrally managed policy control point (that is, a router) that enforces university policy on network traffic. Thus, devices can be connected to switches on the same building access network. The protected systems must also be connected to switches on the same building access network. Devices are placed on the "inside" or "department" side of the switch. Static routes (set centrally) are used to enable the installation of firewalls. Further details on firewall installation are detailed in internal policy documents. Unfortunately, referencing the documents would reveal the identity of the institution. This is further discussed below in the implementation of a firewall for the security department.

When the IDS sensors were delivered, one was initially placed at the Internet handoff for the University, which delivers data at gigabit rates. The device was installed using the Switched Port Analyzer (SPAN) capability. According to Cisco, the primary network vendor:

> A SPAN session is an association of a destination port with a set of source ports, configured with parameters that specify the monitored network traffic. … SPAN sessions do not interfere with the normal operation of the switches. [CISCO 1].

The IDS was rated at 100 MBPS throughput and was woefully undersized for the task, as was the Windows NT 4.0 monitoring station. The monitoring station would crash frequently and could not handle more than a few hours of logs. Additional disk drives and memory were installed. The number of active signatures was strictly limited to those matching a current virus outbreak. Scripts were written to automatically rotate and compress the log files several times a day.

However, responsibility for the project was split with another division. The IDS sensor was under the control of a telecommunications analyst (TA) and the monitoring station by the security analyst (SA). Neither party was allowed to touch or access the device allocated to the other party. Less than ten IDS signatures were permitted. Among them were signatures for Nimda and Code Red, which were active during that time frame. The IDS tended to be fragile, so the TA spent long hours re-installing the Solaris x86 operating system multiple times. Security staff checked log files daily and followed up on the virus-infected systems that appeared in the reports. The project was highest priority for the SA, in part due to the fact that the security division was newly founded and needed to prove itself. At the same time, it was among the lowest of priorities for the TA, who was involved in the ongoing network transition.

The specifications of the network backbone infrastructure were at that time, and remain to the present day, closely held within the Network Operations Group

(NOG). This is especially true of the configuration of the Internet handoff and the configuration of the firewall. Providing an accurate network diagram that fully depicts the network installation is not possible, neither would public dissemination be permitted if the information was known outside of the NOG. The general layout is shown in Figure 2.

The monitoring station, a Windows NT 4.0 server, collected logs on a scheduled basis from the Cisco IDS sniffing device. Unrouted private addresses in the 192.168 (uncoordinated) private address range were assigned to one interface on the monitoring Windows NT server, as well as the monitoring interface on the sensor, depicted in the diagram as a direct connection between the devices primarily for the sake of simplicity. The second interface on the sniffer is in promiscuous mode and snooped to network traffic. Since the feed was at a much higher rate than the sensor was rated to handle, an unknown percentage of packets were most likely dropped. Since so few signatures were allowed, there was no need to fine tune the operation of the IDS further.

Packets that matched the IDS signatures were captured for reporting to the monitoring station. This enabled the identification of hosts purportedly infected with Code Red and Nimda viruses. The persons responsible for the hosts were contacted by the security staff and advised as to how to remove the virus (and other remediation as required). The second interface on the monitoring station connected it to the LAN and permitted Internet access for the host. This was necessary to obtain vendor signature updates, vendor software patches and for remote access to the reports via the web.

It was necessary for vendor patches and signature updates to be applied to the monitoring station by the SA, and then to the sensor by the TA. Since only a few approved signatures were permitted, staying current with new signatures was not an issue. The number of alerts were thus artificially limited to manageable levels.

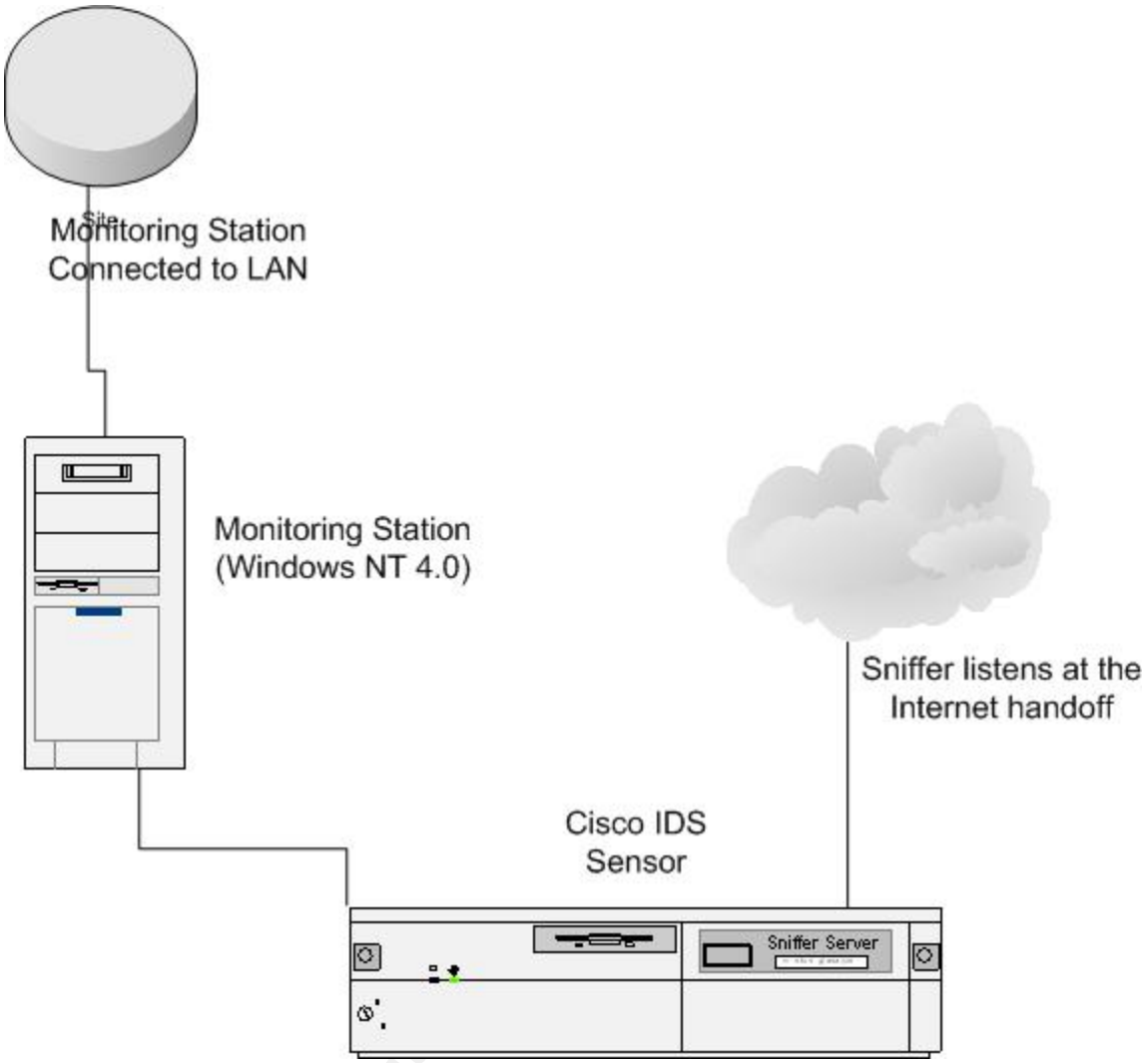


Figure 2: Depiction of IDS installation at the Internet handoff

In this milieu, misunderstanding and frustration blossomed freely. The network administrators seemed less than comfortable with the ability of the security analysts to access records of network activity, while the security analysts were dissatisfied with the small number of signatures that were permitted. As Nimda and Code Red waned, an agreement was reached to gradually expand the number of signatures on the IDS. Unfortunately, management at the director level was not adequately advised of this decision by mid-level management. Following this misunderstanding, the sensor was disabled and removed from the network without prior notice. The immediate reason given was that the SA had increased the installed signature base without permission and without notification, confirming correspondence notwithstanding. Thus the endeavor was declared finished.

At a later time, it was indicated that there were problems with the SPAN on the router at the internet handoff. It was thought to be causing problems with the overall network connectivity. This information was not shared until several months after the fact. The initial picture looked like failure, loss of face and a prime example of the impact of poor communications on project implementation. Given the network constraints and the fact that a third party controlled the departmental LAN used by the security group, a re-deployment plan for the IDS sensors was neither obvious nor imminent.

Solutions

The story continued some months later. The Klez virus hit RESNET (networked dormitories) and spread rapidly via email. The university firewall was configured to provide NAT (Network Address Translation) from private address space for RESNET (Internet access in student dormitory rooms). Discussion and definition of Network Address Translation is beyond the scope of this paper but Cisco provides an excellent online introduction [CISCO 9]. See RFC 1918 for a complete description of private address space. [REKHTER].

While it blocked incoming access for many of the well known ports [IANA], the firewall did not otherwise control or log network access. The bottom line was that RESNET hosts appeared to the outside world with addresses in the University address space, but tedious lookups through high volume logs were required to identify the actual hosts responsible for any Acceptable Use Policy violations. The security staff did not have access to the netflow logs. Inaccurate time stamps further hampered correct identification of miscreant hosts.

The logs primarily used for host identification for Network Address Translation (NAT) hosts in the RESNET are netflow logs. "Cisco IOS NetFlow technology is an integral part of Cisco IOS Software that collects and measures data as it enters specific routers or switch interfaces. By analyzing NetFlow data, a network manager can identify the cause of congestion; determine the class of service (CoS) for each user and application; and identify the source and destination network for your traffic." [CISCO 10] Although intended for traffic analysis, the logs identify source and destination and provide the primary means of identifying hosts which access the Internet via Network Address Translation (NAT).
The timestamp problem was further exacerbated as RESNET hosts were permitted to use outside ISPs for outgoing SMTP service. A locally popular national ISP permitted outgoing SMTP service from RESNET hosts. As the outside ISP obscured the mail handling infrastructure, the email headers of the Klez-infected emails did not match the hosts logged in the netflow logs. There was no clear way to identify the inside hosts from the exterior addresses. The inside (RESNET) hosts were registered in DNS on coordinated private address space in the ranges 172.16.0.0 to 172.27.255.255. Network Address Translation (NAT) assigned a single address for all of the residents of a particular dorm. A specific exterior address and accurate time stamp would permit identification of

the hosts from the netflow logs (kept for 10 days due to high volume). Again, due to the obfuscation by the outside ISP, it was impossible to determine the source of the Klez virus from the email headers and netflow logs alone.

A discussion on how to solve this problem led to an inspirational suggestion. For reasons of practicality, the staff responsible for RESNET had access to and directly managed their switch infrastructure. Further, they were anxious to deal with the virus outbreak as expeditiously as possible. A signature for Klez was available for the IDS, so it seemed like a workable possibility. When approached about the possibility of installing a sensor, they welcomed the chance. A port SPAN was still considered risky, so a TAP [EINWECHTER, NETOPTICS 2, NETOPTICS 3, NETOPTICS 4] was recommended instead. The operating system on the monitoring station was upgraded (gratefully) to Windows 2000 and hardened as per current best practices and vendor recommendations [MICROSOFT], taking advantage of security templates available on the from Microsoft on their web site. A fiber-to-copper TAP (Test Access Port) was ordered, the sensor was moved to a secure remote location in the RESNET switching infrastructure and the TAP was installed within a few days of delivery. The network diagram in Figure 3 illustrates the topology of the redeployment. Due to the passive nature of the TAP device, network connectivity would not be affected even by the loss of power to the TAP (or other hardware failure of the TAP). Traffic continued to move to RESNET over the fiber cable, and was at the same time captured by the IDS sensor.

When all was in place, the sensor appeared to be seeing traffic, but the remote monitoring station did not log the traffic. The large (1548 bytes instead of 1500) packets characteristic of VLAN (virtual local area network) traffic could not be handled by the default interface. On the Cisco web site, a VLAN is defined as

> "…a group of devices on one or more LANs that are configured (using management software) so that they can communicate as if they were attached to the same wire, when in fact they are located on a number of different LAN segments. …VLANs are based on logical instead of physical connections…" [CISCO 12]

A call to Cisco technical support found the solution in a new driver that would enable the interface to handle the larger packet sizes [CISCO 6]. In order to handle this, the sniffer interface became the control (monitoring) interface, and vice versa. Subsequently, the sensor worked smoothly and began logging signature hits viewable from the remote monitoring station.

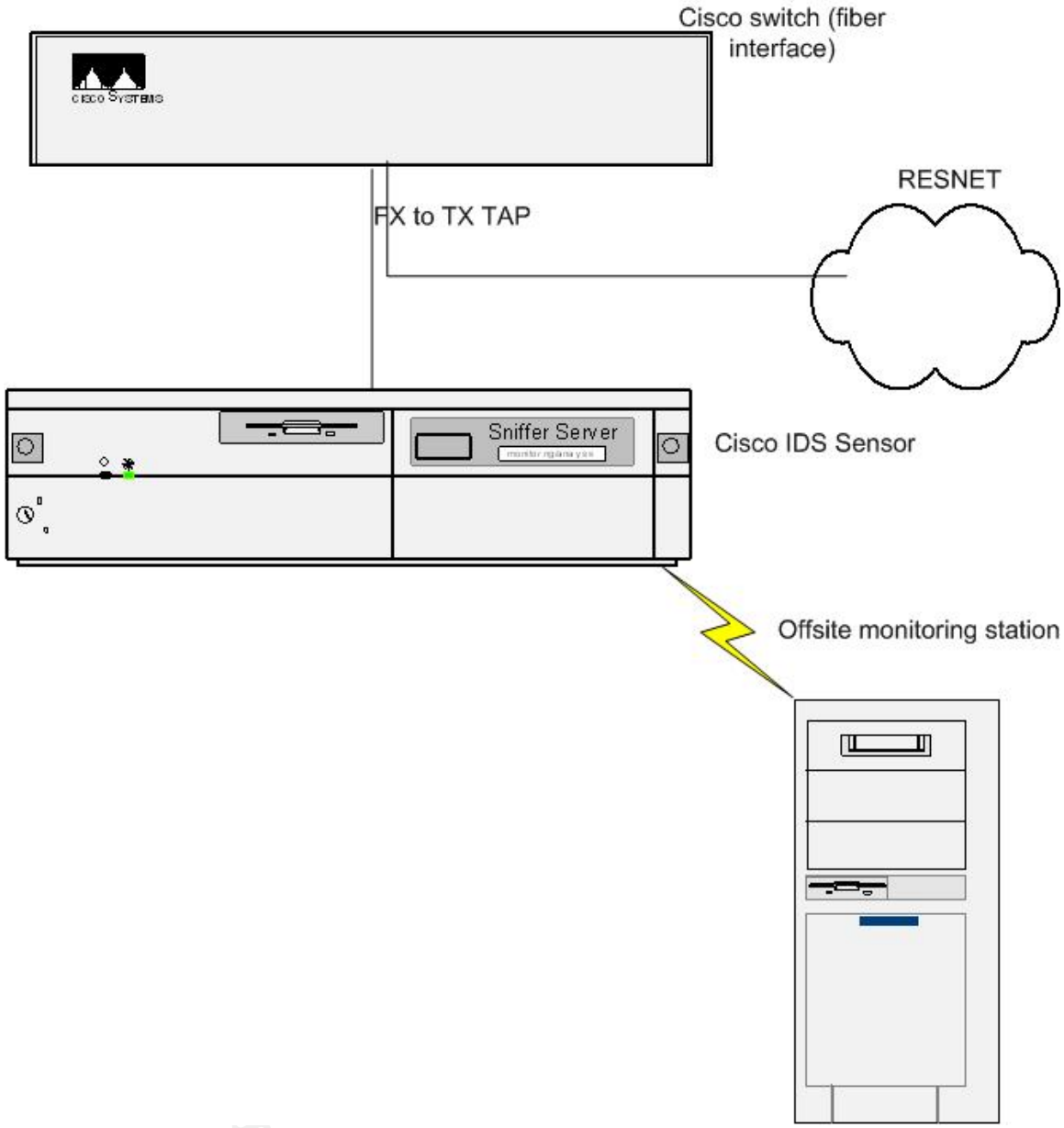


Figure 3: Installation of Cisco IDS sensor in RESNET

During the next semester break period, the fiber TAP was removed and a port SPAN was enabled. Network traffic was light but no problems were noted. As the port SPAN can be modified in the switch configuration software remotely to change the target of the IDS to a different subnet, it would be a preferred approach, pending acquisition of a specialized fiber cable.

Another IDS sensor was plugged into an eight port hub within the security group LAN. Traffic was generated by a cast off workstation that was taken off the hardware maintenance contract but kept in service for this purpose. This allowed experimentation with the device as well as the opportunity to install software upgrades and signature updates on a local test bed before deployment to the

remote RESNET site (for which physical access is a worthy challenge). The signature update procedure is fully documented by the vendor for new set of signatures. These instructions plus the ability to first upgrade a physically available sensor led to a high degree of confidence in the process of updating signatures remotely.

Maintenance of the Cisco IDS involves regular installation of new IDS signatures, as they are published, and examination of the signature hits reported to the monitoring station. It should be further noted that the sensor located in the RESNET monitored one floor of one residence hall. Although all signatures were now turned on, including one for Klez, the number of alerts was still low. Further, the alerts are ranked and color coded by the vendor as informational, low, medium or high depending on the perceived severity of the logged attack. Staff simply concentrated on the alerts ranked as medium or high. This was easier since the alert viewing software color coded the alerts with red, orange, yellow and blue to correspond to high, medium, low and informational messages. Datagrams (not even the headers) were not available for further analysis; only the results presented by the management interface. Again, the amount of traffic was limited. While normally uninteresting signatures should be turned off, they were left on to verify that the device was still working. As physical access required 3 separate keys (and coordination with RESNET staff) to enter a fortress-like basement area, this was a non-trivial concern.

Interesting traffic was further reduced as anti-virus appliances were installed to scan incoming and outgoing email on the central servers. The immediate effect was to significantly reduce the number of email borne virus infections in the university as a whole, although users employing outside SMTP servers still sent copies of Klez. The ISP generously made enough information available about its internal mail infrastructure that two endpoints were identifiable. Along with the time stamp, infected NAT hosts then could be reliably identified through the netflow logs. Current plans are to SPAN the switch so that the target subnet for the IDS can be changed in software. Another possibility is to redeploy the IDS to another department, or possibly behind the recently installed firewall in the security department LAN.

The departmental firewall for the security staff has been recently installed. The general topology is illustrated in Figure 4. The Cisco IDS previously installed mainly as a test bed for signature installations and software upgrades will now be put to more effective use, as illustrated in the proposal as shown in Figure 5. A further advantage is found in that direct and instructive comparisons between the Cisco IDS and Snort IDS can be made, as they will be monitoring the same network traffic. It is anticipated that this will provide a hitherto unavailable opportunity to test the reliability of the IDS devices against each other. Further, it will be possible to examine the raw traffic when deemed necessary or interesting. Since logs of network traffic are generally unavailable outside of the division that supports the network infrastructure, this is the most likely means of determining

reliability. Requests for information are entertained in connection with specific instances of computer abuse. Source IP, target IP and time stamp are required. Further note that if the general topology is found to be successful, a switch will replace the hub. Either a copper TAP or SPAN on the switch will permit at least one of the IDS devices to be connected to the network.

Given previous and painful experience, written plans and network diagrams were developed and subject to technical review by experts from two outside divisional groups as well as the Chief Technology Officer. Implementation is expected to begin shortly. Future plans include a failover firewall, and possibly the deployment of a second Cisco IDS on the inside of the firewall to examine traffic that passes through the firewall.

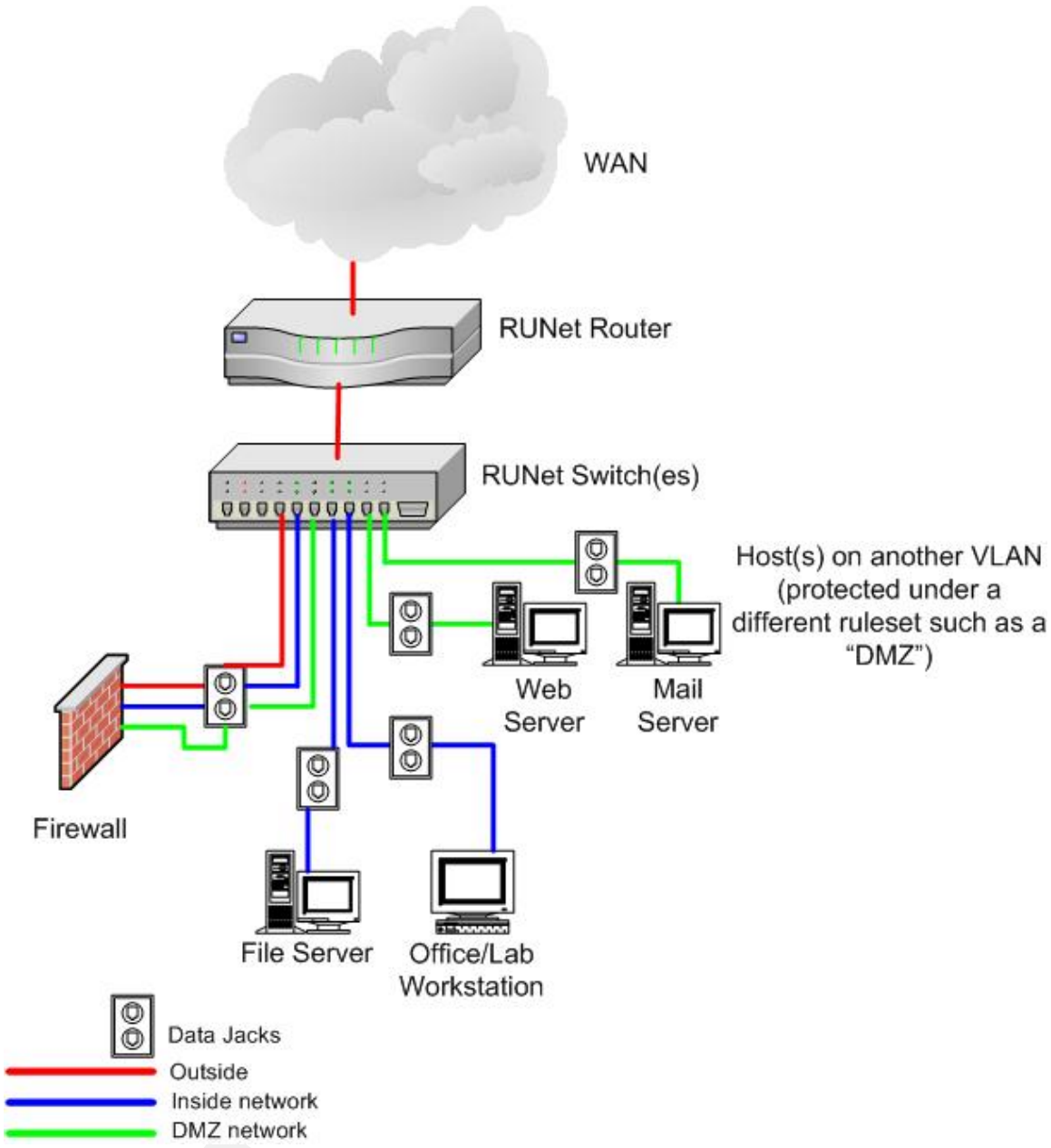


Figure 4: Single firewall protecting single network with DMZ network

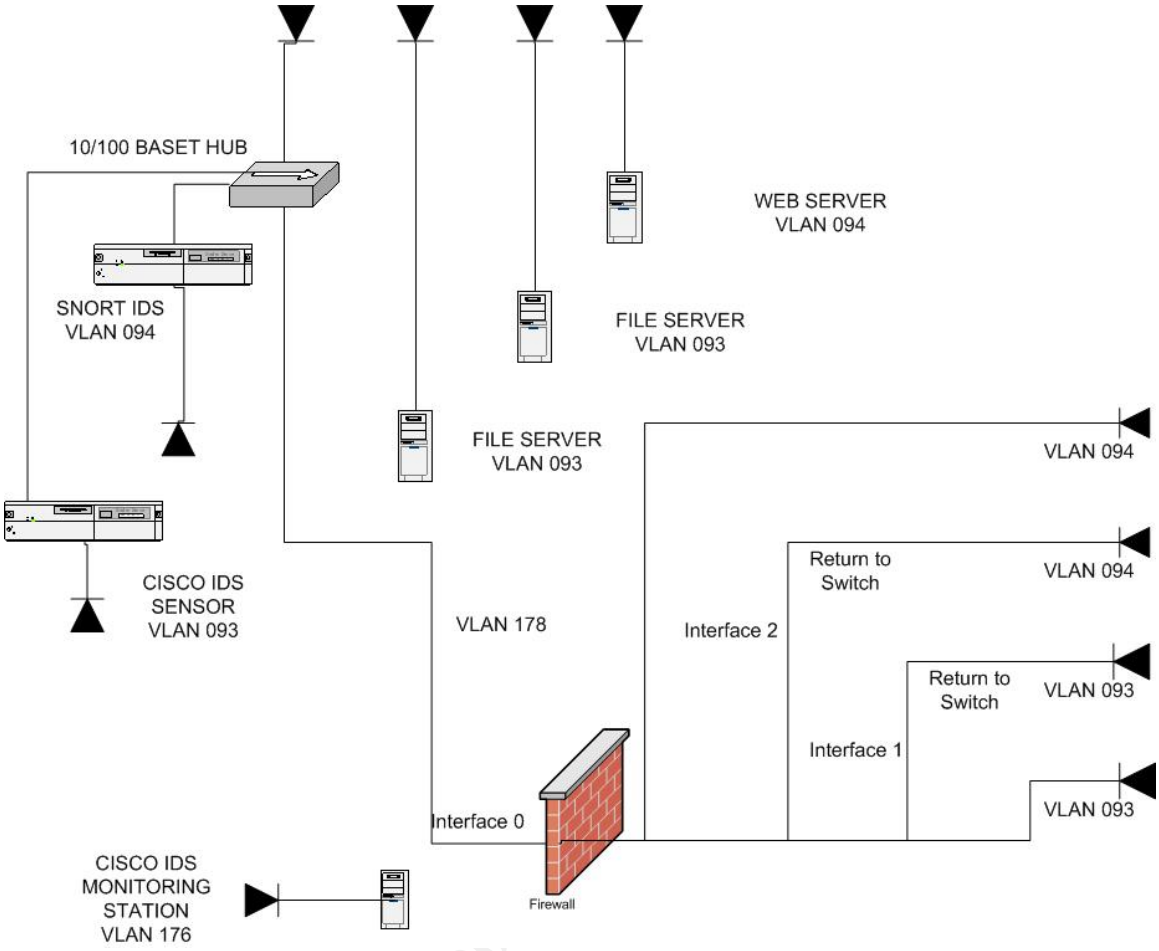


Figure 5: Proposed LAN Diagram (pending implementation)

Conclusions and Recommendations

One of the challenges in the early days was the lack of experience with Windows and Cisco hardware and software. Procedures have been documented to cover:

- IDS sensor installation
- signature update
- Windows 2000 installation
- Windows 2000 hardening procedures
- hardware and software inventory for the project

The “lessons learned” are requirements for any comparable future project, and for any large scale project:

- Develop a formal written project plan
- Obtain written approval of the project plan from upper management

- Specify the expectations, controls, and responsibilities of all participants in a joint project
- Train staff in IDS and network technologies prior to the onset of the project

Management of the IDS technology was significantly improved by access to the sensor hardware and software as well as the monitoring device. In the first iteration, the security staff had no experience with the sensor device and was barred from acquiring the important "hands on" experience in the technology. In retrospect, killing off the first project was the best thing that could have happened, although admittedly it did not appear so at the time.

These further initiatives have been identified as projects that will enhance the overall security posture:

- Port 25 (SMTP) traffic from RESNET will be restricted to central email servers with virus checking software. This will significantly impact the propagation of viruses and worms spread via email
- Enable port SPAN for RESNET IDS while it remains in a RESNET location
- Configure and install both IDS sensors for security group LAN (possible now that the firewall is installed)

Some time has passed since the inception of the IDS project. In the interval, upper management recognized the need for project management training across the IT organization and made available an opportunity for online study. Although it was a hard lesson, it was also highly effective. While it was not free, it certainly could have been far more expensive in terms of dollars, time and effort. At present, the IDS sensors are running in two locations and plans for redeployment look promising.

References

- [CISCO 1] Cisco Systems, Inc. http://www.cisco.com/univercd/cc/td/doc/product/iaabu/csids/csids8/13877_01.htm (March 8, 2003).

- [CISCO 2] "Cisco Intrusion Detection System Getting Started Version 3.1" Cisco Systems, Inc. http://www.cisco.com/univercd/cc/td/doc/product/iaabu/csids/csids8/13872_01.htm (March 8, 2003).

- [CISCO 3] "Cisco Intrusion Detection System Sensor Configuration Note Version 3.1" Cisco Systems, Inc. http://www.cisco.com/univercd/cc/td/doc/product/iaabu/csids/csids8/13870_01.htm (March 8, 2003)

- [CISCO 4] "Cisco Catalyst 6500 Series Switches" Cisco Systems, Inc. http://www.cisco.com/en/US/products/hw/switches/ps708/products_configuration_guide_chapter09186a008007f323.html Jan 18, 2003 (March 8, 2003).

- [CISCO 5] "Cisco Intrusion Detection System Device Manager Configuration Note Version 3.1." Cisco Systems, Inc. http://www.cisco.com/univercd/cc/td/doc/product/iaabu/csids/csids8/13876_01.htm (March 8, 2003).

- [CISCO 6] "Cisco Intrusion Detection System Event Viewer Configuration Note Version 3.1." h 8, 2003).

- [CISCO 7] "Cisco Intrusion Detection System Sensor Version 3.0 VLAN Trunking Instructions". San Jose: Cisco Systems, Inc. September, 2001.

- [CISCO 8] "Cisco Intrusion Detection System Signature Engines Version 3.1" Cisco Systems, Inc. http://www.cisco.com/univercd/cc/td/doc/product/iaabu/csids/csids8/1386_01.htm (March 8, 2003).

- [CISCO 9] "NAT: Local and Global Definitions" Cisco Systems, Inc. http://www.cisco.com/warp/public/556/8.html (March 8, 2003).

- [CISCO 10] "Netflow" Cisco Systems, Inc. http://www.cisco.com/en/US/tech/tk648/tk362/tk812/tech_protocol_home.html (March 8, 2003)

- [CISCO 11] Release Notes for Cisco Intrusion Detection System Sensor Version 3.1. Cisco Systems, Inc.

http://www.cisco.com/univercd/cc/td/doc/product/iaabu/csids/csids8/13871_01.htm (March 8, 2003).

- [CISCO 12] "VLAN". Cisco Systems, Inc. http://www.cisco.com/en/US/tech/tk389/tk689/tech_protocol_family_home.html (March 8, 2003).

- [EINWCHTER] Einwechter, Nathan. "Implementing Networks Taps with Network Intrusion Detection Systems**".** Security Focus http://www.securityfocus.com/infocus/1594 (March 8, 2003).

- [NETOPTICS 1] "The Case for TAPS". Net Optics. http://www.netoptics.com/case1.html (March 8, 2003).

- [NETOPTICS 2] "FX Tap to TX". Net Optics. http://www.netoptics.com/fx-tx-tap.html (March 8, 2003).

- [NETOPTICS 2] "FX to TX Tap" Net Optics. http://www.netoptics.com/fx-tx-tap.pdf (March 8, 2003).

- [NETOPTICS 3] "Installation Guide: FX to TX Tap." Net Optics. http://www.netoptics.com/install-fx-tx-tap.pdf (March 8, 2003).

- [IANA] IANA (Internet Assigned Numbers Authority), "Port Numbers", http://www.iana.org/assignments/port-numbers March 6, 2003 (March 8, 2003).

- [MICHAEL] Michael, Martin J. Understanding the Network: A Practical Guide to Internetworking. Indianapolis, IN: New Riders. 2000.

- [MICROSOFT] Windows 2000 Server Baseline Security Checklist. Redmond: Microsoft Corporation. http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/chklist/w2ksvrcl.asp (March 8, 2003).

- [REKHTER] Rekhter, Y.; Moskowitz, B.; Karrenberg, D; de Groot, G. J.; Lear, E "Address Allocation for Private Internets" ftp://ftp.rfc-editor.org/in-notes/rfc1918.txt February, 1996 (March 8, 2003).

- [SUN] Sun Microsystems System Administration Guide, Volume I. Palo Alto: Sun Microsystems. http://docs-pdf.sun.com/805-3727/805-3727.pdf (March 8, 2003).

Assignment 2: Network Detects

Detect 1: This detect was posted to intrusions@incidents.org on March 1, 2003. The original and follow up postings and can be viewed here:

http://cert.uni-stuttgart.de/archive/intrusions/2003/03/msg00006.html
http://cert.uni-stuttgart.de/archive/intrusions/2003/03/msg00008.html
http://cert.uni-stuttgart.de/archive/intrusions/2003/03/msg00035.html
http://cert.uni-stuttgart.de/archive/intrusions/2003/03/msg00042.html

Top 3 questions or comments on the original detect:

1. Initial statement:

The primary attacking IP address is 216.232.36.98, which accounted for 225 of the generated alerts for hits on port 1080. There is a minimun of 10 seconds between each packet, sometimes longer. It may be an attempt to drop "below the radar" or perhaps indicates that this scan is part of a larger one involving other networks as well.

Question/comment:

Which one would You [sic] guess? I think that's an important part of the analysis. I see some clues in the packets You [sic] listed that might point to one interpretation and not the other.

Rebuttal:

The breaks in sequence numbers and the time delays between the packets tend to indicate that this scan is part of a larger scan involving other networks. It is also possible that it is an attempt to obfuscate the scan. A much slower scan would be far more effective so if that is the motive, the attacker is not accomplishing the goal well.

2. Initial statement:

As the packets are TCP packets with the SYN flag only set, the attack has the characterics of a stimulus, hoping for a response on port 1080. Possible targets are the SOCKS proxy service, an installed WinHole Trojan or SubSeven 2.2 trojan. Since the TTL of the attacker is consistently 113, the attacker would more likely than not be running Windows.

Question/comment:

Maybe, but do You [sic] see evidence of packet crafting? If the packet is crafted, You [sic] may not be able to make this statement.

Rebuttal:

If the host is a "newer" Windows host, the default initial TTL is 128 (according to http://project.honeynet.org/papers/finger/traces.txt [SPITZNER]) indicating that the packets have taken 15 hops for each detect, which seems to be on the high side for number of hops. Windows cannot be entirely ruled out, though.

The following lines from tcpdump give further indication of packet crafting:

```
19:00:49.414488 216.232.36.98.23698 > 78.37.179.26.1080: S
288666012:288666012(0) win 512 <mss 1460,eol> (DF)

19:01:18.514488 216.232.36.98.2134 > 78.37.185.26.1080:
S 288793500:288793500(0) win 512 <mss 1460,eol> (DF)

19:02:02.074488 216.232.36.98.34297 > 78.37.194.26.1080:
S 288984732:288984732(0) win 512 <mss 1460,eol> (DF)

19:02:21.394488 216.232.36.98.19921 > 78.37.198.26.1080:
S 289069724:289069724(0) win 512 <mss 1460,eol> (DF)
```

The TCP option EOL is used to "pad" TCP options that don't fall on a byte boundary. However, an mss option of 1460 without any other options would not require padding. The use of this TCP option when it is manifestly unneeded points towards packet crafting.

3.

Initial Statement:

The port report at Dshield Distributed Intrusion Detection System at http://dshield.org [DSHIELD] provided contemporary correlation (February, 2003) of the abundance of port 1080 probes. At this writing, it ranks in the "top 20" of scanned ports.

Question/commentary:

Does DShield list attacks from this IP address? Is it known to be an SOCKS attacker? Is it maybe scanning for other Windows backdoors on other hosts? That might tell You [sic] if the scan really is for SOCKS or for some backdoor, as You [sic] suggest might be the case above.

Rebuttal:

Historical data was not available from Dshield, but MyNetWatchman had a report on the same IP address from mid-May, 2002, which would

be around the same time frame. Browse to this location for details: http://mynetwatchman.com/LID.asp?IID=4531678. The same host also probed port 80 and ports 8080 and 3128 (RingZero) in the same time frame.

Given the target port of 1080, several reporting sites flagged the activity as SOCKS proxy scanning:

MyNetWatchman
http://mynetwatchman.com/kb/security/ports/6/1080.htm

ISS
http://www.iss.net/security_center/advice/Intrusions/2003017/default.htm

Robert Graham
http://www.robertgraham.com/pubs/firewall-seen.html#port1080

While Subseven could run on port 1080, it is apparently more commonly associated with other ports, as is WinHole. See http://www.robertgraham.com/pubs/firewall-seen.html#subseven and http://www.anti-trojan.net/en/trojportlist.aspx. The attack is most likely associated with SOCKS open proxy scanning. The remediation should focus on proxy servers first, and then scan for the SubSeven and Winhole trojans.

1.) Source of Trace:

This detect came from the raw tcpdump logs available on incidents.org at the following URL:

http://www.incidents.org/logs/Raw/2002.4.15

2.) Detect was generated by:

The detect was generated by running the following command on the captured traffic using Snort 1.9 (installed on a Sun Ultra 10 running Solaris 2.7) and a recent (February 24, 2003) ruleset:

```
snort -c /etc/snort.conf -r 2002.4.14 -l $HOME/log.2002.4.15
```

The Snort rule that triggered the detect has Signature ID 615 [SNORT]:

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 1080 (msg:"SCAN SOCKS Proxy
attempt"; flags:S,12; reference:url,help.undernet.org/proxyscan/;
classtype:attempted-recon; sid:615; rev:4;)
```

The rule triggers on external accesses to port 1080.

The resulting data was analyzed with several tools. The summaries provided by SnortSnarf were an excellent means of gaining a broad overview of the data while Ethereal version 0.9.9 (installed on a Dell OptiPlex GX100 running Windows 2000 Professional) provided an easy means for detailed examination of individual packets of interest. The tcpdump program allows direct examination of the data sans froufrou.

Several different attacks are in evidence in this log file.
Of sample files available at http://www.incidents.org/logs/Raw/
for possible analysis, the file 2002.4.15 ranks 33 out of 140 in terms of file size, and accounts for the richness of detect. Snort triggered on six different signatures, displayed via SnortSnarf as follows:

Signature	# Alerts	# Sources	# Destinations
Portscan detected	3	1	3
BAD TRAFFIC ip reserved bit set	1	1	1
BAD TRAFFIC bad frag bits	10	7	8
DNS named version attempt	54	9	54
SCAN nmap TCP	94	13	33
SCAN SOCKS Proxy attempt	237	6	227

While all attacks are of some interest, attention is drawn to the SCAN SOCKS Proxy attempt, with a high number of alerts, a similarly high number of destinations, and a relatively low number of sources. Three Snort alerts are illustrative:

```
[**] [1:615:3] SCAN SOCKS Proxy attempt [**]
 [Classification: Attempted Information Leak] [Priority: 2]
 05/14-19:00:49.414488 216.232.36.98:23698 ->
78.37.179.26:1080
 TCP TTL:113 TOS:0x0 ID:1168 IpLen:20 DgmLen:48 DF
 ******S* Seq: 0x1134B19C Ack: 0x0 Win: 0x200 TcpLen: 28
 TCP Options (2) => MSS: 1460 EOL
 [Xref => url help.undernet.org/proxyscan/]

[**] [1:615:3] SCAN SOCKS Proxy attempt [**]
 [Classification: Attempted Information Leak] [Priority: 2]
 05/14-19:01:18.514488 216.232.36.98:2134 ->
78.37.185.26:1080
 TCP TTL:113 TOS:0x0 ID:1168 IpLen:20 DgmLen:48 DF
 ******S* Seq: 0x1136A39C Ack: 0x0 Win: 0x200 TcpLen: 28
 TCP Options (2) => MSS: 1460 EOL
 [Xref => url help.undernet.org/proxyscan/]
```

```
[**] [1:615:3] SCAN SOCKS Proxy attempt [**]
 [Classification: Attempted Information Leak] [Priority: 2]
 05/14-19:02:21.394488 216.232.36.98:19921 ->
78.37.198.26:1080
 TCP TTL:113 TOS:0x0 ID:1168 IpLen:20 DgmLen:48 DF
 ******S* Seq: 0x113ADA9C Ack: 0x0 Win: 0x200 TcpLen: 28
 TCP Options (2) => MSS: 1460 EOL
 [Xref => url help.undernet.org/proxyscan/]
```

The target is probably a Class B network (78.37.*.*). The true subnets have been disguised to appear as part of an IANA reserved address range [IANA 1]. The attacker is allegedly from TELUS Communications, a regional ISP in western Canada. Using my private (non-work) ISP connection for a lookup uncovers the exact hostname, *amhe6t3y30ci.bc.hsia.telus.net* and a browse of the ISP web site (http://www.telus.net) seems to indicate that the IP address is associated with ADSL service located in British Columbia. Without further information, it is unknown as to whether the IP address is dynamically or statically assigned to the user, but it is more likely a home user or SOHO (Small Office/Home Office), and as Microsoft is so prevalent in that market, there is a somewhat higher chance that the host runs a Microsoft Windows operating system.

Running tcpdump on the log file shows 456 responses from the host 78.37.212.28, all to port 80. The web sites would appear to be unrelated to work duties for most employees in the US. Examples include hitbox.com, personalfinance.aol.com, and several sites devoted to home building. An email message that was routed through yahoo.com can best be described as explicitly recreational in nature. Perhaps the target site employs custom signatures that look for certain phrases that are typical of “non-duty” web surfing.

3.) Probability the source address was spoofed:

Since the attack looks like a reconnaissance on port 1080, the source address is unlikely to be spoofed. The attacker will want to get the results back in order to launch further attacks or take advantage of vulnerabilities associated with port 1080 in order to attack or use the proxy to mask the actual source of other attacks. A attacker controlling several hosts on the Internet might launch the attack from host 1, spoof the address as host 2 (also controlled by the attacker) and collect the results at host 2. This would indicate a higher level of sophistication and so is less likely than a straightforward attack. There is also the possibility that the attacker has already penetrated the network and is able to sniff the response as it comes across the wire.

4.) Description of attack:

The primary attacking IP address is 216.232.36.98, which accounted for 225 of the generated alerts for hits on port 1080. There is a minimum of 10 seconds

between each packet, sometimes longer. It may be an attempt to drop "below the radar" or perhaps indicates that this scan is part of a larger one involving other networks as well. The breaks in sequence numbers and the time delays between the packets tend to indicate that this scan is part of a larger scan involving other networks. It is also possible that it is an attempt to obfuscate the scan. A much slower scan would be far more effective so if that is the motive, the attacker is not accomplishing the goal well. Possible targets (in addition to the SOCKS proxy service) include an installed WinHole Trojan or SubSeven 2.2 trojan.

5.) Attack mechanism:

As the packets are TCP packets with the SYN flag only set, the attack has the characteristics of a stimulus, hoping for a response on port 1080. Since the TTL of the attacker is consistently 113, the attacker would more likely than not be running Windows.

The following lines from tcpdump give further indication of packet crafting:

```
19:00:49.414488 216.232.36.98.23698 > 78.37.179.26.1080: S
288666012:288666012(0) win 512 <mss 1460,eol> (DF)

19:01:18.514488 216.232.36.98.2134 > 78.37.185.26.1080:
S 288793500:288793500(0) win 512 <mss 1460,eol> (DF)

19:02:02.074488 216.232.36.98.34297 > 78.37.194.26.1080:
S 288984732:288984732(0) win 512 <mss 1460,eol> (DF)

19:02:21.394488 216.232.36.98.19921 > 78.37.198.26.1080:
S 289069724:289069724(0) win 512 <mss 1460,eol> (DF)
```

The TCP option EOL is used to "pad" TCP options that don't fall on a byte boundary. However, an mss option of 1460 without any other options would not require padding. The odd TCP option point towards likely packet crafting. The use of the DF (“don’t fragment”) also seems unusual for a datagram of 1460. Fragmentation comes into play when datagrams exceed the standard Ethernet MTU (1500). Normally, if the host is a "newer" Windows host, the default initial TTL is 128, indicating that the packets have taken 15 hops for each detect, which seems to be on the high side for number of hops. [SPITZNER]. Since other parts of the packets show evidence of crafting, the validity of the TTL is more suspect, the relative market share of Microsoft notwithstanding.

A web search uncovered tools and tutorials on proxy scanning, of which two are presented.

Proxybench
http://www.proxybench.com/proxy/checker.asp

Proxybench features the following capabilities:

- Finds SOCKS proxies (SOCKS4 and SOCKS5)
- Ability to scan a range of IPs
- Ability to test specific IP addresses from a list
- Multi-threaded for high speed scanning
- Sends a test email to confirm proxy can mail
- SOCKS Proxy Scanner Checker Validator

SOCKS proxies are specifically recommended for accessing Internet Relay Chat (IRC) anonymously, although it is also pointed out that some IRC servers check for the use of a proxy server. In general, it will permit anonymous access of the Internet and the launching of multiple other kinds of attacks. Another tutorial oriented towards the tool Proxyhunter can be found at http://www.jestrix.net/tuts/scan.html. The writer recommends using one of the "free" ISP services since such scanning might cause loss of access privileges.

6.) Correlations:

The port report at Dshield Distributed Intrusion Detection System at http://dshield.org provided contemporary correlation (February, 2003) of the abundance of port 1080 probes. At this writing, it ranks in the "top 20" of scanned ports.

Historical data was not available from Dshield, but MyNetWatchman had a report on the same IP address from mid-May of 2002, which would be around the same time frame as the scan examined in this detect. See http://mynetwatchman.com/LID.asp?IID=4531678 for details on the activity. The same host also probed port 80 and ports 8080 and 3128 (RingZero) at about the same time frame. Further, several sites noted that port 1080 is associated with scanning for open SOCKS proxies [ISS, MYNETWATCHMAN, GRAHAM].

7.) Evidence of active targeting:

Loading the 2002.4.15 log file into Ethereal provides the opportunity for further data analysis. Sorting by Source IP address indicates a pattern in the destination IP address. Hosts ending in .26 were attacked first, then .27, .28 and so on up to .36. There is a predictable pattern to the destination IP addresses in the scan. The nature of the attack is more consistent with reconnaissance than with a concerted attach on one specific host so it would not be characterized as active targeting.

8.) Severity:

Given the following formula:

severity = (criticality + lethality) - (system countermeasures + network countermeasures) where 1 is lowest and 5 the highest value for each variable, the severity of this attack as follows:

Criticality = 3

The functions of the various systems are unknown; a medium value was chosen.
Lethality = 4

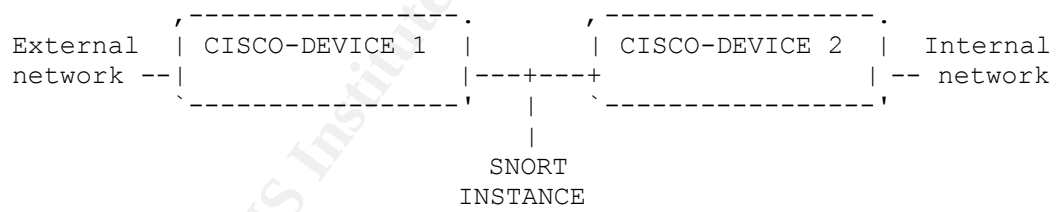
Discovering an unprotected proxy is a juicy target. It can be used to mask attacks against other network hosts, making it difficult to trace back to the origin. Further, there are buffer overflow vulnerabilities associated with SOCKS. [SF1, SF 2, ETHEREAL, SF3, SF4, SF5, SF6, SF7].

System countermeasures = 3

An IDS is in place and there is evidence of custom signatures, so this level of care would indicate that it is more (rather than less) likely that system countermeasures are in place, although there is no direct evidence.

Network countermeasures = 4

An IDS is in place and there is evidence that custom signatures are installed, so the level of care is probably somewhat higher. As with Andre Cormier's analysis [CORMIER], the MAC address for both source and destination refer to Cisco devices [ETHEREAL], so the topology probably resembles his diagram, reproduced here from his excellent 20 January 2003 posting:



No reliable inferences about which ports may be blocked, so the assumption will be made that traffic is allowed to targets.

(3 + 4) - (3 + 4) gives an overall rating of 0 for this attack.

9.) Defensive recommendation:

- Remove SOCKS service if it is unneeded.
- If SOCKS service is required, upgrade to current versions and secure it as per vendor recommendations. In general, the proxy server needs to be

configured so that users inside the network can access it, but users from the outside cannot come back in. Of critical importance is configuration of the Local Address Table, which must correctly list internal address ranges. [COOPERS].

- Use anti-virus software to actively search for Subseven and Winhole trojans. Verify that the AV software and signatures are kept current.

10.) Multiple choice question:

```
19:00:49.414488 216.232.36.98.23698 > 78.37.179.26.1080: S
288666012:288666012(0) win 512 <mss 1460,eol> (DF)
19:01:18.514488 216.232.36.98.2134 > 78.37.185.26.1080:
S 288793500:288793500(0) win 512 <mss 1460,eol> (DF)

19:02:02.074488 216.232.36.98.34297 > 78.37.194.26.1080:
S 288984732:288984732(0) win 512 <mss 1460,eol> (DF)

19:02:21.394488 216.232.36.98.19921 > 78.37.198.26.1080:
S 289069724:289069724(0) win 512 <mss 1460,eol> (DF)
```

In the trace above, EOL signifies:

a) end of list
b) a TCP option
c) end of life
d) end of line

Correct answer: b

EOL is a TCP option indicating that pad bytes were needed before the next option could start. [STEVENS (page 93)]

References

- [CVE] Common Vulnerabilities and Exposures http://www.cve.mitre.org/ (March 8, 2003).

- [COOPERS] Coopers & Lybrand: Microsoft Proxy Server Security Evaluation http://msdn.microsoft.com/library/en-us/dnproxy/html/msdn_proxycase.asp (March 8, 2003).

- [CORMIER] Cormier, A., "LOGS: GIAC GCIA Version 3.3 Practical Detect(s) (Andre Cormier)" 20 January 2003, intrusions@incidents.org, archive by Universitt Stuttgart's CERT, <http://cert.uni-stuttgart.de/archive/intrusions/2003/01/msg00162.html> (6 March 2003).

- [DSHIELD] Dshield Port Report http://www.dshield.org/port_report.php (March 8, 2003).

- [ETHEREAL] Ethereal http://www.ethereal.com/ (March 8, 2003).

- [GRAHAM] Graham, Robert http://www.robertgraham.com/pubs/firewall-seen.html#port1080 (March 8, 2003).

- [IANA 1] “IP Option Numbers” Internet Assigned Numbers Authority http://www.iana.org/assignments/ip-parameters March 6, 2001 (March 8, 2003).

- [IANA 2] Port Numbers http://www.iana.org/assignments/port-numbers (March 8, 2003).

- [ISS] ISS http://www.iss.net/security_center/advice/Intrusions/2003017/default.htm (March 8, 2003)

- [MYNETWATCHMAN] MyNetWatchman. http://mynetwatchman.com/kb/security/ports/6/1080.htm (March 8, 2003).

- [PORT] Port Lookup Utility http://www.treachery.net/security_tools/ports/ (March 8, 2003).

- [SANS] Intrusion Detection FAQ: What port numbers do well-known trojan horses use? http://www.sans.org/resources/idfaq/oddports.php (March 8, 2003)

- [SD] SnortSnarf http://www.silicondefense.com/software/snortsnarf/ (March 8, 2003).

- [SF 1] AN HTTPD Malformed SOCKS4 Request Buffer Overflow Vulnerability (Vulnerabilities) http://www.securityfocus.com/bid/6012 (March 8, 2003).

- [SF2] AnalogX Proxy Socks4A Buffer Overflow Vulnerability (Vulnerabilities) http://www.securityfocus.com/bid/5138 (March 8, 2003).

- [SF 3] Ethereal SOCKS Dissector Memory Corruption Vulnerability (Vulnerabilities) http://www.securityfocus.com/bid/5163 (March 8, 2003).

- [SF 4] NEC Socks5 User Name Buffer Overflow Vulnerability (Vulnerabilities) http://www.securityfocus.com/bid/5145 (March 8, 2003).

- [SF 5] NEC Socks4 User Name Buffer Overflow Vulnerability (Vulnerabilities) http://www.securityfocus.com/bid/5147 (March 8, 2003).

- [SF 6] NEC Socks5 Host Name Off-By-One Buffer Overflow Vulnerability (Vulnerabilities) http://www.securityfocus.com/bid/5149 (March 8, 2003).

- [SF 7] Socks5 1.0r5 Buffer Overflow Vulnerability (Vulnerabilities) http://www.securityfocus.com/bid/154 (March 8, 2003).

- [SF 8] T. Hauck Jana Server SOCKS5 Proxy Server Authentication Buffer Overflow Vulnerability (Vulnerabilities) http://www.securityfocus.com/bid/5321 July 26, 2002 (March 8, 2003)

- [SNORT] Snort http://www.snort.org/ (March 8, 2003).

- [SPADE] Sam Spade http://www.samspade.org (March 8, 2003).

- [SPITZNER] Spitzner, Lance. Lists of fingerprints for passive fingerprint monitoring http://project.honeynet.org/papers/finger/traces.txt (March 8, 2003).

- [STEVENS] Stevens, W. Richard, TCP/IP Illusatrated, Volume 1: The Protocols Reading: Addison-Wesley Publishing Co, 1994.

- [TCPDUMP] tcpdump http://tcpdump.org/ (March 8, 2003).

- [UNDERNET] Undernet Scans for Insecure Wingates and Proxies http://help.undernet.org/proxyscan/ (March 8, 2003).

- [WHITEHATS] Whitehats Port Query http://www.whitehats.ca/main/tools/portquery2/portquery2.html (March 8, 2003).

Detect 2: This detect was posted to intrusions@incidents.org on March 5, 2003. The original and follow up postings and can be viewed here:

http://cert.uni-stuttgart.de/archive/intrusions/2003/03/msg00077.html
http://cert.uni-stuttgart.de/archive/intrusions/2003/03/msg00079.html
http://cert.uni-stuttgart.de/archive/intrusions/2003/03/msg00094.html
http://cert.uni-stuttgart.de/archive/intrusions/2003/03/msg00082.html
http://cert.uni-stuttgart.de/archive/intrusions/2003/03/msg00080.html
http://cert.uni-stuttgart.de/archive/intrusions/2003/03/msg00081.html
http://cert.uni-stuttgart.de/archive/intrusions/2003/03/msg00083.html

1. Source of Trace:

This detect came from the raw tcpdump logs available on incidents.org at the following URL:

http://www.incidents.org/logs/Raw/2002.4.16

2. Detect was generated by:

The detect was generated by running the following command on the captured traffic using Snort 1.9.1 [SNORT] (installed on a Sun Ultra 10 running Solaris 2.7) with a recent (February 24, 2003) ruleset:

snort -c /etc/snort.conf -r 2002.4.16 -l $HOME/log.2002.4.16

The resulting data was analyzed with SnortSnarf [SD], Ethereal [ETHEREAL] version 0.9.9 (installed on a Dell OptiPlex GX100 running Windows 2000 Professional) and [TCPDUMP].

SnortSnarf was run with the following command:

./snortsnarf.pl -d $HOME/tmp/2002.4.16/ -rulesfile /etc/
$HOME/log.2002.4.16/alert

As expected, several different attacks are in evidence. This detect will focus on the DNS named version attempts. The spread of alerts is depicted in the table on the next page. The particular signature (1616) that triggered the alarm is:

```
alert udp $EXTERNAL_NET any -> $HOME_NET 53 (msg:"DNS named version
attempt"; content:"|07|version"; nocase; offset:12; content:"|04|bind";
nocase; offset: 12; reference:nessus,10028; reference:arachnids,278;
classtype:attempted-recon; sid:1616; rev:4;)
```

The rule triggers on external access to port 53.

Signature	# Alerts	# Sources	# Destinations
BAD TRAFFIC bad frag bits	9	7	7
SCAN Squid Proxy attempt	21	3	4
SCAN Proxy (8080) attempt	21	3	4
SCAN SOCKS Proxy attempt	39	4	5
DNS named version attempt	45	10	45
SCAN nmap TCP	47	11	12

Output of sample packets via Snort:

```
[**] [1:1616:4] DNS named version attempt [**]
[Classification: Attempted Information Leak] [Priority: 2]
05/16-05:02:58.364488 203.155.236.133:2770 -> 78.37.101.18:53
UDP TTL:44 TOS:0x0 ID:64725 IpLen:20 DgmLen:58
Len: 38
[Xref => arachnids 278][Xref => nessus 10028]

[**] [1:1616:4] DNS named version attempt [**]
[Classification: Attempted Information Leak] [Priority: 2]
05/16-05:09:57.704488 203.122.47.137:12177 -> 78.37.174.6:53
UDP TTL:41 TOS:0x0 ID:39994 IpLen:20 DgmLen:58
Len: 38
[Xref => arachnids 278][Xref => nessus 10028]

[**] [1:1616:4] DNS named version attempt [**]
[Classification: Attempted Information Leak] [Priority: 2]
05/16-05:27:14.164488 203.122.47.137:28740 -> 78.37.24.197:53
UDP TTL:41 TOS:0x0 ID:59150 IpLen:20 DgmLen:58
Len: 38
[Xref => arachnids 278][Xref => nessus 10028]
```

Output of (same) sample packets via tcpdump (truncated to fit on this page):

```
05:02:58.364488 203.155.236.133.2770 > 78.37.101.18.53:  4660
[b2&3=0x80] TXT CHAOS? version.bind. (30)

05:09:57.704488 203.122.47.137.12177 > 78.37.174.6.53:  4660
[b2&3=0x80] TXT CHAOS? version.bind. (30)

05:27:14.164488 203.122.47.137.28740 > 78.37.24.197.53:  4660
[b2&3=0x80] TXT CHAOS? version.bind. (30)
```

Given the total spread of addresses, the target is most likely a class B network, although the target IP addresses are known to be sanitized.

3. Probability the source address was spoofed:

The Whitehats [WHITEHATS] site opines that since it is a UDP packet that causes this event (DNS is a UDP based service) the source address could be easily forged. However, this scan would be pointless if the information was not returnable to the attacker. Whitehats also judges it as less likely that the attack was spoofed for this reason. There is also the possibility that the attacker has already penetrated the network and is able to sniff the response as it comes across the wire. Since the three way handshake is not necessary for UDP traffic, this tends to raise the likelihood of spoofing.

4. Description of attack:

The attack seeks to determine the DNS version, reconnaissance for later attacks on the DNS servers that are part of the subnet. The attack pattern is interesting as each inside host was probed exactly once, but the probes come from 10 different sources and might indicate that the attack was coordinated among several hostile and cooperating hosts. A more prosaic possibility is simply that the hacker already has access to a number of different hosts to use as launching points for attacks. The attack incorporates a mostly obsolete query, chaos, to request the version number, which would support the presumption that old and vulnerable versions of BIND were sought by the attacker.

The version of BIND can be queried on the command line with nslookup as well as dig. [FREE]:

```
nslookup -q=txt -class=CHAOS version.bind. 0

dig @ducky.nz.freebsd.org version.bind chaos txt
```

The local DNS server would be substituted for "@ducky.nz.freebsd.org". As these queries are not accepted by the DNS servers here, it was not possible to fully verify that the produce the requested information. The server noted that it could not find "version.bind" and that the query was refused.

If the version can be determined, the payoff is rich. Over 30 vulnerabilities are listed in the Common Vulnerabilities and Exposure [CVE] database, most of them including specific BIND version numbers. A short sampling includes cache poisoning, denial of service attacks, execution of arbitrary code by the attackers, unauthorized dynamic updates of DNS, and the gaining of root privileges. Further, once the DNS server is "owned" the entire site can be controlled. More BIND vulnerabilities are documented at the Internet Software Consortium site [ISC]. The attack is considered significant enough to be included in the SANS/FBI Top 20 [SANS 2].

5. Attack mechanism:

The attack is a stimulus hoping for a response that will reveal the version of DNS running on the server, if it is a indeed a DNS server. Freely available scanners check for BIND version number; among them are Nessus [NESSUS] and [SARA]. The scanners can easily be directed towards specific IP address ranges. It is to be noted that the destination addresses typically reported only one signature hit. At the same time, there are several different source addresses. It would appear that scans are launched from multiple locations at the same time.

There is a rich variety of exploits and vulnerabilities based on different version numbers of BIND. For example, the “nxt bug” [CERT] could allow an intruder to oferflow a buffer and execute arbitrary code with the root privileges. Details and exploit code are available [SF 1]. The successful attacker will gain a remote shell with root privileges. This could be one of several attacks that could be launched based on successful reconnaissance.

A possible attack tool is the Linux/Lion worm [MCAFEE]. It uses a random port scan to seek systems that contain a root access vulnerability in the BIND DNS service on Linux servers. Once a target is found, the system is attempted for compromise, and password information is sent to an email address in China. A variant of Lion uses the bind exploit to infect the system, sets up to listen on port port 27374 and feeds it a web page. It sends an email to China with the /etc/passwd and /etc/shadow files. Running the files through a password cracker will eventually yield usernames and passwords. Lion generates class B network addresses and uses pscan scan random class B internet address space. [SANS 3]. Pscan (http://www.insecure.org/nmap/scanners/pscan.c) styles itself as a TCP/UDP/NIS/RPC scanner that

- scans TCP ports and prints the services running
- scans UDP ports and prints the services running (remote hosts only)
- dumps portmappers listing of RPC services
- prints available NIS maps

It would appear to be an extremely useful tool for network reconnaissance.

The “ADM w0rm” was discussed on the Bugtraq mailing list in March of 1999 as another Internet worm that scans the hosts and exploits BIND vulnerabilities. Since recent material was not readily at hand, it is mentioned only in passing here. [SF 2]

6. Correlations:

In order to provide a summary of the data, the attacking hosts are grouped by country. In Thailand and Turkey, the hosts were from the same ISP. In India,

two organizations are represented, a technical institute in Bombay being predominant. None presently appear as attackers in the Dshield [DSHEILD 2] databases. MyNetwatchman [MYNETWATCHMAN] has history records of many DNS scans coming from 203.122.47.137, and one from 203.197.101.55 as follows:

203.155.237.173	Thailand (x)	
203.155.236.133	Thailand (x)	
203.122.47.137	India	Multiple DNS scan reports
203.197.102.32	India (@)	
203.197.102.43	India (@)	
203.197.101.55	India (@)	DNS scan reports
217.131.173.179	Turkey (*)	
217.131.191.70	Turkey (*)	
217.131.175.127	Turkey (*)	
217.131.174.4	Turkey (*)	

The Dshield site [DSHIELD] ranks port 53 as among the top 30 destination ports as an attack vector. It did not make the top current rankings on the MyNetwatchman site [MYNETWATCHMAN].

7. Evidence of active targeting:

This appears to be a general scan with a motive of reconnaissance; active targeting will come later. The top three attackers are from the following IP addresses:

Address	Country of Origin
203.155.237.173	Thailand
203.122.47.137.1	India
217.131.173.179	Turkey

Scanning for this vulnerability is extremely prolific. However, it is the results of these scans that will be used later in the directed attacks to gain root shells, launch denial of service attacks and other mischief.

8. Severity:

Formula: severity = (criticality + lethality) - (system countermeasures + network countermeasures) where 1 is lowest and 5 highest value for each variable, the severity of this attack as follows:

Criticality = 5

A DNS server is a critical target [SANS 1, page 9].

Lethality = 1

The scan itself does no harm; it is preparation for a more serious attack.

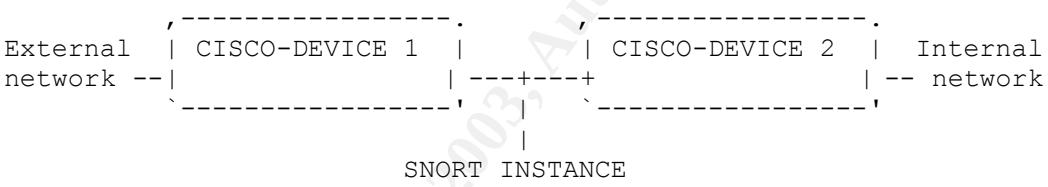
System Countermeasures = 2

The system countermeasure in place are unknown, so the ranking will be somewhat pessimistic.

Network Countermeasures = 2

The site is running an IDS and this attack was detected, so there are at least some countermeasures in place.

Cormier's analysis in his 20 January 2003 posting points out that the MAC address for both source and destination refer to Cisco devices [ETHEREAL] and infers that the topology looks like his diagram, reproduced here from his posting [CORMIER]:



No reliable inferences about which ports may be blocked, so the assumption will be made that traffic is allowed to targets.

(5 + 1) - (2 + 2) = 2

Overall ranking of 2 for this attack.

9. Defensive recommendation:

- Don't run BIND (or any service) unless it is explicitly needed.

- Upgrade to the latest version of BIND and/or apply any vendor patches. The Internet Software Consortium provides an open version of BIND software. (http://www.isc.org/products/BIND/)

- Set up "split horizon" DNS to limit information available to outsiders.

- Remove versioning information from BIND.

- Follow other "best practices" such as restricting zone transfers and dynamic updates.

Specific references on securing DNS/BIND:

Secure BIND Template v3.7
http://www.cymru.com/Documents/secure-bind-template.html

Securing an Internet Name Server
http://www.acmebw.com/resources/papers/securing.pdf

10. Multiple choice question:

Hackers scan for the BIND (DNS) version for the following reason:

a) because they wish to prove to other hackers that they can.

b) DNS servers are not a target of interest

c) because they like to gather any and all information about a possible target host even if inconsequential

d) primarily as reconnaissance, later to target DNS servers with attacks that will be effective.

Correct answer: (d)

References:

- [CERT] CERT Advisory CA-1999-14 Multiple Vulnerabilities in BIND http://www.cert.org/advisories/CA-1999-14.html (March 10, 2003).

- [CVE] Common Vulnerabilities and Exposures http://www.cve.mitre.org/ (March 8, 2003).

- [CORMIER] Cormier, A., "LOGS: GIAC GCIA Version 3.3 Practical Detect(s) (Andre Cormier)" 20 January 2003, intrusions@incidents.org, archive by Universitt Stuttgart's CERT, <http://cert.uni-stuttgart.de/archive/intrusions/2003/01/msg00162.html> (6 March 2003).

- [DSHIELD 1] Dshield Port Report http://www.dshield.org/port_report.php (March 8, 2003).

- [DSHIELD 2] Dshield Port 53 – DNS http://www.dshield.org/ports/port53.php (March 8, 2003).

- [ETHEREAL] Ethereal http://www.ethereal.com/ (March 8, 2003).

- [FREE] “What version of BIND are you running?” http://www.freebsddiary.org/bind-version.php (May 10, 2003)

- [IANA] Port Numbers http://www.iana.org/assignments/port-numbers (March 8, 2003).

- [ISC] “BIND vulnerabilities” Internet Software Consortium http://www.isc.org/products/BIND/bind-security.html (March 8, 2003).

- [LIU] Liu, Cricket. Securing an Internet Name Server http://www.acmebw.com/resources/papers/securing.pdf (March 8, 2003).

- [MCAFEE] Linux/Lion Worm. http://vil.nai.com/vil/content/v_99056.htm Virus Information Library (May 10, 2003).

- [MYNETWATCHMAN] MyNetWatchman http://www.mynetwatchman.com/ (March 8, 2003).

- [NESSUS] Nessus. http://www.nessus.org (March 8, 2003)

- [SANS 1] "Network Based Intrusion Detection Tutorial 1" IDS Signatures and Analysis, Parts 1 and 2. SANS Institute. 2002.

- [SANS 2] “SANS/FBI Top 20 List” http://www.sans.org SANS Institute. Version 3.22. March 3, 2003. (May 11, 2003).

- [SANS 3] “Lion Worm” http://www.sans.org/y2k/lion.htm SANS Institute. Version 0.12 April 18, 2001 (May 11, 2003).

- [SARA] Security Auditor’s Research Assistant (SARA). Center for Internet Security http://www.cisecurity.org (March 8, 2003).

- [SD] SnortSnarf http://www.silicondefense.com/software/snortsnarf/ (March 8, 2003).

- [SF 1] “Multiple Vendor BIND (NXT Overflow and Denial of Service Vulnerabilities” http://securityfocus.com/bid/788/exploit (May 11, 2003).

- [SF 2] “Re: ADM Worm. Worm for Linux x86 found in wild” http://securityfocus.com/archive/1/12995 Bugtraq archive. March 26, 1999. (May 11, 2003).

- [SNORT] Snort http://www.snort.org/ (March 8, 2003).

- [SPITZNER] Spitzner, Lance. Lists of fingerprints for passive fingerprint monitoring http://project.honeynet.org/papers/finger/traces.txt (March 8, 2003).

- [TCPDUMP] tcpdump http://tcpdump.org/ (March 8, 2003).

- [THOMAS] Thomas, Rob. Secure BIND Template v3.7 http://www.cymru.com/Documents/secure-bind-template.html (March 8, 2003).

- [WHITEHATS] Named Probe http://www.whitehats.com/info/IDS278 (March 8, 2003).

Detect 3:

1. Source of Trace:

This detect came from the raw tcpdump logs available on incidents.org at the following URL:

http://www.incidents.org/logs/Raw/2002.5.20

2. Detect was generated by:

Running the following command on the captured traffic using Snort 1.9.1 [SNORT] (installed on a Sun Ultra 10 running Solaris 2.7) with a recent (February 24, 2003) default ruleset:

```
snort -c /etc/snort.conf -r 2002.5.20 -l $HOME/log.2002.5.20
```

The resulting data was analyzed with SnortSnarf [SD] , Ethereal [ETHEREAL] version 0.9.9 (installed on a Dell OptiPlex GX100 running Windows 2000 Professional) and tcpdump [TCPDUMP].

SnortSnarf command:

```
./snortsnarf.pl -d $HOME/tmp/2002.5.20 -rulesfile /etc/
$HOME/log.2002.5.20/alert
```

The summary of attacks from SnortSnarf [SD] is presented below, followed by Snort output. The SCAN FIN alerts were chosen as the focus of interest.

Signature	**# Alerts**	**# Sources**	**# Destinations**
BAD TRAFFIC bad frag bits	1	1	1
BACKDOOR Q access	47	1	47
SCAN FIN	2	2	1
SCAN Squid Proxy attempt	2	2	1
SCAN Proxy (8080) attempt	2	2	1
SCAN nmap TCP	4	3	2
DNS named version attempt	34	8	34
SCAN SOCKS Proxy attempt	40	3	4
SHELLCODE x86 inc ebx NOOP	48	21	1
SHELLCODE x86 NOOP	51	11	1

The signature that was triggered is SID 621 [SNORT]:

```
alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg:"SCAN FIN";
flags:F,12; reference:arachnids,27; classtype:attempted-recon; sid:621;
rev:2;)
```

The rule triggers on traffic with the FIN and SCAN flags both set.

The two Snort alerts that were generated:

```
[**] [1:621:1] SCAN FIN [**]
[Classification: Attempted Information Leak] [Priority: 2]
06/20-05:36:52.874488 217.208.42.220:20000 -> 46.5.218.182:6346
TCP TTL:46 TOS:0x0 ID:45072 IpLen:20 DgmLen:40 DF
*******F Seq: 0x2E7ED165  Ack: 0x0  Win: 0xFFFF  TcpLen: 20
[Xref => arachnids 27]

[**] [1:621:1] SCAN FIN [**]
[Classification: Attempted Information Leak] [Priority: 2]
06/20-05:54:21.464488 12.253.150.137:61579 -> 46.5.218.182:6346
TCP TTL:44 TOS:0x0 ID:4937 IpLen:20 DgmLen:40 DF
*******F Seq: 0x8F4EF138  Ack: 0x0  Win: 0x4000  TcpLen: 20
[Xref => arachnids 27]
```

Looking at the data again to focus on the destination address with this tcpdump command:

tcpdump -n -r 2002.5.20 dst 46.5.218.182

netted an interesting result.

```
05:36:52.874488 217.208.42.220.20000 > 46.5.218.182.6346: F
780063077:780063077(0) win 65535 (DF)
05:54:21.464488 12.253.150.137.61579 > 46.5.218.182.6346: F
2404315448:2404315448(0) win 16384 (DF)
05:57:49.484488 148.63.151.214.2856 > 46.5.218.182.6346: P
3546041866:3546042036(170) win 8192 (DF)
05:59:50.734488 148.63.151.214.2856 > 46.5.218.182.6346: P
3546041866:3546042036(170) win 8192 (DF)
06:02:00.974488 148.63.151.214.2856 > 46.5.218.182.6346: P
3546041866:3546042036(170) win 8192 (DF)
06:03:26.474488 148.63.151.214.2856 > 46.5.218.182.6346: P
3546041866:3546042036(170) win 8192 (DF)
06:04:07.914488 148.63.151.214.2856 > 46.5.218.182.6346: P
3546041866:3546042036(170) win 8192 (DF)
06:05:10.914488 148.63.151.214.2856 > 46.5.218.182.6346: P
3546041866:3546042036(170) win 8192 (DF)
06:06:45.194488 148.63.151.214.2856 > 46.5.218.182.6346: P
3546041866:3546042036(170) win 8192 (DF)
06:07:20.724488 148.63.151.214.2856 > 46.5.218.182.6346: P
3546041866:3546042036(170) win 8192 (DF)
```

Turning on hex showed this, typical of the 8 that were generated:

```
06:07:20.724488 148.63.151.214.2856 > 46.5.218.182.6346: P
3546041866:3546042036(170) win 8192 (DF)

0x0000   4500 00d2 2eab 4000 6d06 afaf 943f 97d6       E.....@.m....?..
0x0010   2e05 dab6 0b28 18ca d35c 4e0a 0000 0000       .....(...\N.....
0x0020   5e08 2000 a104 0000 474e 5554 454c 4c41       ^.......GNUTELLA
0x0030   2043 4f4e 4e45 4354 2f30 2e36 0d0a 5573       .CONNECT/0.6..Us
0x0040   6572 2d41 6765 6e74 3a20 4265 6172 5368       erAgent:.BearSh
0x0050   6172 6520 322e 362e 330d 0a4d 6163 6869      are.2.6.3..Machi
0x0060   6e65 3a20 312c 382c 3235 352c 312c 3830      ne:.1,8,255,1,80
0x0070   390d 0a50 6f6e 672d 4361 6368 696e 673a       9..PongCaching:
0x0080   2030 2e31 0d0a 486f 7073 2d46 6c6f 773a       .0.1..HopsFlow:
0x0090   2031 2e30 0d0a 4c69 7374 656e 2d49 503a       .1.0..ListenIP:
0x00a0   2031 3438 2e36 332e 3135 312e 3231 343a      .148.63.151.214:
0x00b0   3633 3436 0d0a 5265 6d6f 7465 2d49 503a       6346..RemoteIP:
0x00c0   2031 3730 2e31 3239 2e38 382e 3532 0d0a      .170.129.88.52..
0x00d0   0d0a                                          ..
```

The outside host is attempted to accessing a peer-to-peer (P2P) file sharing service, allegedly running BearShare [BEARSHARE]. The actual IP address of the target host seems as if it might be 170.129.88.52 (not the obfuscated 46.5.180.250 seems to be in the IANA reserved range [SPADE]). The apparent "real" address is owned by

OrgName: Standard Microsystems Corporation
OrgID: SMC-9
Address: 300 Kennedy Drive
City: Hauppauge Industrial Park
StateProv: NY
PostalCode:
Country: US

The company web site (http://www.smsc.com/) indicates that the company is a semiconductor manufacture.

The attacking host seems to be:

OrgName: Spacenet, Inc.
OrgID: SPAN
Address: 1750 Old Meadow Rd
City: Mclean
StateProv: VA
PostalCode: 22102-4300
Country: US

The company sells very small aperture terminal (VSAT) satellite technology and has an informative web site at http://www.spacenet.com .

One "FIN" packet came from AT&T WorldNet Services and the other from Telia Network Services, an ISP based in Sweden.

Some time was invested in determining why Snort did not flag the gnutella traffic with an alert. It was finally determined that the standard rule set did not include the signature for an incoming gnutella request (SID 559), only for outgoing requests.

3. Probability the source address was spoofed:

The source IP address could be quite easily forged, since it is not thought to be part of an existing TCP session. However, since this is an information gathering probe, the aspiring intruder is less likely to have spoofed the source address.

Follow up probes requesting files via BearShare, a Peer-to-Peer file sharing service, so the host address for 46.5.180.250 in particular seems unlikely to have been spoofed.

4. Description of attack:

The FIN scan is used in stealth port scanning to determine if a given port is active or not. A listening port should not respond, while a port that is not listening would respond with a RESET/ACK. This constitutes an indirect means of determining which ports are open. Although there are only two probes, both are for port 6346. The Whitehats site [WHITEHATS 1, WHITEHATS 2] associates port 6346 with Gnutella and gnutella file sharing. Apparently the attacker is searching for possible file sharing sites. This is consistent with the attempts to access files from the same server a few minutes after the FIN packets were sent.

5. Attack mechanism:

This attack is a reconnaissance for active ports running the gnutella protocol, directly followed by an attempt to access the host using a gnutella client. It is inferred that the FIN scan showed port 6346 to be active and one attacker followed with an attempt to download files. The attack could easily be accomplish with nmap, which permits a FIN scan to be generated. Since there are only two probes and they are from different, apparently unrelated sites, and over ten minutes apart, this would not appear to be part of a coordinated attack.

Since the two identified sites are both high tech companies, the possibility of industrial espionage should also be investigated. Although it may be remote it would be prudent to investigate and rule it out.

6. Correlations:

The following log files show attempts to access host 46.5.218.182 on port 6346. This would tend to confirm that the host is actively sharing files to the Internet over a period of about 2 months.

2002.5.4
2002.5.5
2002.5.6
2002.5.7
2002.5.14
2002.5.19
2002.5.21
2002.5.24
2002.5.25
2002.6.9
2002.6.12
2002.6.17
2002.6.18

The general pattern in the other logs was 2 FIN probes, followed by several accesses to port 6346 (gnutella).

Dshield [DSHIELD] did not have confirming reports about 148.63.151.214, 217.208.42.220, or 12.253.150.137. MyNetWatchman [MYNETWATCHMAN] flagged 217.208.42.220 for gnutella probes in the June, 2002 timeframe, which correlates with the data that we have. The address 12.253.150.137 logged incidents but they don't appear to be related to gnutella.

7. Evidence of active targeting:

There is a possibility that although the accesses are from 3 different hosts, the attack is coordinated, as several attempts to access files follow the port 6346 scan in short order. This attack is directed toward the host 46.5.218.182 and that host alone.

8. Severity:

Given the following formula: severity = (criticality + lethality) - (system countermeasures + network countermeasures) where 1 is lowest and 5 highest value for each variable, the severity of this attack is calculated as follows:

Criticality = 3

The function of the victim host is unknown beyond its file sharing activity. Presumably it has other IT function in the corporation. It may or may not have critical files, so the median value is assigned.

Lethality = 3

The lethality of the attack depends on what files are being downloaded from the file server, whether they represent sensitive and "company confidential" information or copyrighted entertainment material (videos, songs, et cetera). In the first instance, the damage to the company might be more extensive. In the absence of hard information, the median value was chosen.

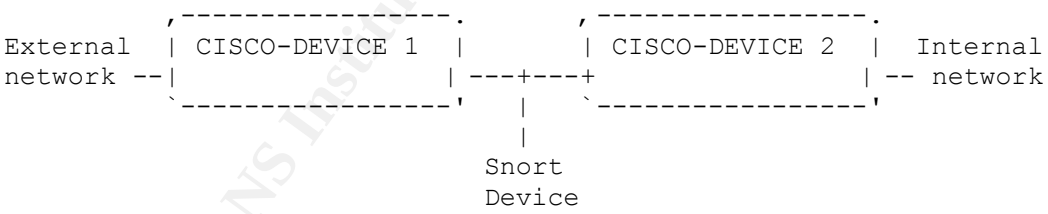
System countermeasures = 3

Unknown, so the median value is chosen

Network countermeasures = 3

The target site is running an IDS and detected the activity, so there are some countermeasures in place. Further, it would seem that "extra" signatures are turned on that are not part of the default Snort rule set. As signature #559 is turned on to check for outgoing gnutella requests, it would seem likely that running P2P servers may be a violation of the corporate acceptable use policy.

Again referring to Cormier's analysis of 20 January 2003 [CORMIER], the MAC address for both source and destination [ETHEREAL] refer to Cisco devices, implies that the topology looks like his diagram, reproduced here from his posting:



Overall rating: (3 + 3) - (3 + 3) is 0 for this attack.

9. Defensive recommendations:

- Assuming such programs are against corporate policy, perform an internal scan to locate other hosts with file sharing programs.
- Block port 6346 at the firewall (if any).
- Develop (or review) corporate policy on the file sharing programs and the approval process for the installation of software.

- Review copyright law and the Digital Milennium Copyright Act with users.

- Check for business relationships (either cooperative or competitive) with the companies responsible for the probes.

10. Multiple choice question:

The FIN flag in the TCP header:

a) refers to the anatomy of a fish
b) terminates a TCP connection
c) stands for "Format Identification Number"
d) initiates a TCP connection

The correct answer is (b) The F stands for
FIN, which terminates a connection. [NOVAK 2, page 2-28].

References:

- [BEARSHARE] BearShare. http://www.bearshare.com/ (5 March 2003).
- [CORMIER] Cormier, A., "LOGS: GIAC GCIA Version 3.3 Practical Detect(s) (Andre Cormier)" 20 January 2003, intrusions@incidents.org, archive by Universitt Stuttgart's CERT, http://cert.uni-stuttgart.de/archive/intrusions/2003/01/msg00162.html (6 March 2003).
- [DSHIELD] Dshield. http://dshield.org (5 March 2003).
- [ETHEREAL] Ethereal. http://www.ethereal.com/ (5 March 2003).
- [IANA 1] Port Numbers. http://www.iana.org/assignments/port-numbers (5 March 2003).
- [MYNETWATCHMAN] MyNetWatchman. http://mynetwatchman.com (5 March 2003).
- [NOVAK 1] Novak, Judy. "IP Behavior IV: Stimulus --> Response" TCP/IP for Intrusion Detection. SANS Institute, 2002.
- [NOVAK 2] Novak, Judy. "TCP/IP Refresher and Beyond" TCP/IP for Intrusion Detection. SANS Institute, 2002.
- [PORT] Port Lookup. http://www.treachery.net/tools/ports/lookup.cgi (5 March 2003).
- [SD] SnortSnarf http://www.silicondefense.com/software/snortsnarf/ (5 March 2003).
- [SNORT] Snort. http://www.snort.org (5 Marxch 2003).
- [SPADE] Sam Spade http://www.samspade.org (March 8, 2003).
- [TCPDUMP] tcpdump http://tcpdump.org/ (March 8, 2003).
- [WHITEHATS 1] Probe FIN Scan. http://www.whitehats.com/IDS/27 (5 March 2003)
- [WHITEHATS 2] Whitehats Port Query http://www.whitehats.ca/main/tools/portquery2/portquery2.html (5 March 2003).

Assignment 3: "Analyze This"

Overview

This network is in good hands. The network administrators are running the Snort IDS and are well-versed in it, enough to write custom rules. However, the rules should be reviewed on a regular basis to evaluate their continued efficacy. Security would significantly benefit by migration to secure services (e. g., ssh and sftp). The firewall seems to have some unnecessary ports open (69 and 137 were identified); consideration should be given to locking them down. Attention should also be paid to the primary attacking sites, especially hosts within the university, which should normally be easier to remediate than those overseas. For those hosts outside of the local network, attempts to work with the ISP may reduce the number of attacks. Possible vulnerabilities associated with ports 4888 and especially 2561 Mosaix Predictive Dialing System) should be investigated further. Finally, user awareness and education are primary in the security plan of any organization.

Logs Analyzed

The logs were taken from the http://www.incidents.org/logs web site and cover five days, starting Sunday, February 23, 2003 and ending Thursday, February 27, 2003. The logs comprise Snort alerts, scan data, and "Out of Spec" packets (those with strange or illegal combinations of flags set).

Filename	Size
alert.030223.gz	480,269
alert.030224.gz	411,323
alert.030225.gz	427,630
alert.030226.gz	503,789
alert.030227.gz	496,996

In order to detect trends over the period as a whole, the data was combined into a single large file for analysis. This involved changing the delimiters in the file to the common delimiter of % so that it could be imported into the SAS Institute [SAS] statistical program (SAS version 8.2 running on a Solaris 9 platform). The chosen methodology was to focus on those events that appeared most often in

the files. Therefore, the statistical analysis of the data generated one-way frequency tables on the data. This identified the alerts or hosts of interest. Additional invocations of "grep" (the UNIX command for searching for regular expressions within text files) isolated the alerts of interest for any further statistical analysis and more detailed study.

For the genesis of the script to prepare the data for loading into SAS, I am indebted to Brandon Newport's GCIA practical [NEWPORT] for the initial version of the script. My own variant on the initial script follows:

```
grep -v spp_portscan $1 | sed 's/->/\%/g' | sed 's/:/%/3' | sed 's/:/%/4' > $2
```

The alerts are a product of Snort [SNORT], version unknown, with custom rules as well as standard rules that are documented on the Snort web site.

As part of the data preparation, 199 corrupted records were completely removed from the input. The alerts on "Tiny Fragments – Possible Hostile Activity" were extracted and set aside as these alerts did not show source and destination ports. Similarly, the spp_portscan records were skipped. This left 152,414 alerts for processing.

Filename	Size
scans.030223.gz	304,623
scans.030224.gz	107,409
scans.030225.gz	96,186
scans.030226.gz	383,202
scans.030227.gz	141,365

Similarly, the scans were processed in one large file.

Filename	Size
OOS_Report_2003_02_23_22505	1,991,683
OOS_Report_2003_02_24_24091	563,203
OOS_Report_2003_02_25_11706	588,803
OOS_Report_2003_02_26_32018	957,443
OOS_Report_2003_02_27_17540	757,763

The final group of logs are the "Out Of Spec" packets, which are TCP packets with strange or illegal combinations of flags set. Again, they were examined as a whole rather than one day at a time in order to examine a broader base of trends in the data.

Top Alerts

Alerts with a frequency higher than 10,000 were chosen for primary analysis. Together they comprise 80% of the alert events. They are as follows:

Alert	Frequency	Percentage
SMB Name Wildcard	71,725	47.06
Watchlist 000220 IL-ISDNNET-990517	25,715	16.87
spp_http_decode - IIS Unicode attack detected	13,373	8.77
High port 65535 tcp - possible Red Worm - traffic	11,479	7.53

Alert: SMB Name Wildcard
Severity: Critical
Frequency: 71,725

No default rule by this name was found on the Snort web site. However, a web search turns up references to the rule, so it would appears to be a rule that has since been discarded but is still in use at the target University. Server Message Block (SMB) is the protocol that Microsoft uses to share files, printers, and serial ports. SMB is also used to communicate between computers by using named pipes and mail slots. In a networked environment, servers make file systems and resources available to clients. Clients make SMB requests for resources, and servers make SMB responses in what is described as a client server, request-response protocol. [MICROSOFT 1] A further search of the Microsoft site turned up a description of wildcards in SMB, describing the use of special characters ? and * for matching characters. As in UNIX, the asterisk matches an entire part of the file name. [MICROSOFT 2]. From this information, it is inferred that the SMB Name Wildcard alert refers to attempts to access any available file using the SMB protocol. This could indeed be a treasure trove of information and data.

Typical samples of alert hits follow:

02/23-00:46:22.224327 [**] SMB Name Wildcard [**] 203.69.163.222:1026 -> MY.NET.248.113:137
02/23-00:46:22.676903 [**] SMB Name Wildcard [**] 217.225.12.145:1545 -> MY.NET.91.109:137

These detects are to port 137, NETBIOS name service [HAWRYLKIW, MATHIS]. Clever use of the UNIX command grep, cut and sort reveal that there are over 16,000 sources for these attempts, the most prevalent being 67.83.29.116 with

347 accesses and 207.6.57.6 with 226 attempts. The command used to identify the particular alerts in the file:

grep “SMB Name Wildcard” all.data | cut –d% -f4 | sort –u | wc –l

The frequency counts were generated by taking the output of the above grep command as input to SAS for one-way frequency table generation. Similar approaches were used for other event counts.

Since these hosts seem to have an interest in the University network, the registration information for these hosts was examined. Use of Geektools [GEEKTOOLS] shows the ownership of the addresses. The first is a block of network addresses maintained by a large ISP, Optimum Online (Cablevision Systems), which owns the range 67.83.24.0 – 67.83.31.255. The output from a registration lookup on whois.arin.net was sparse but did indicate an association with Cablevision Systems. A web search turned up optonline.net as another good candidate for further information. The registrant for both is CSC Holdings, as shown below:

Registrant:
CSC Holdings, Inc. (OPTONLINE2-DOM)
1111 Stewart Ave.
Bethpage, NY 11714
US

Domain Name: OPTONLINE.NET

Administrative Contact:
eMedia Administrator (VTDCADRGXO) cvdomain@CABLEVISION.COM
eMedia Administrator
1111 STEWART AVE
BETHPAGE, NY 11714-3533
US
516-803-3000 fax: - - 516-803-1186
Technical Contact:
Hostmaster, OOL (APTKWSNRPI) hostmaster@CV.NET

111 New South Road
Hicksville, NY 11801
US
(516)393-3281 (516)390-9439

Registrant:
CSC Holdings, Inc. (CABLEVISION-DOM)
1111 Stewart Avenue

Bethpage, NY 11714
US

Domain Name: CABLEVISION.COM

Administrative Contact:
eMedia Administrator (VTDCADRGXO) cvdomain@CABLEVISION.COM
eMedia Administrator
1111 STEWART AVE
BETHPAGE, NY 11714-3533
US
516-803-3000 fax: - - 516-803-1186
Technical Contact:
Murphy, James (SVRDKUALYI) jmurphy@CABLEVISION.COM

1111 STEWART AVE
BETHPAGE, NY 11714-3533
US
516-803-3871 516-803-3950

The other is a Canadian ISP, TELUS Communications Inc., located in Burnaby, British Columbia. The actual hostnames would appear to indicate that the IP addresses are part of a modem or DSL pool. The registration information follows:

OrgName: TELUS Communications Inc.
OrgID: TACE
Address: #2600 4720 Kingsway Avenue
City: Burnaby
StateProv: BC
PostalCode: V5N-4N2
Country: CA

NetRange: 207.6.0.0 - 207.6.255.255
CIDR: 207.6.0.0/16
NetName: TELUS-207-6-0-0
NetHandle: NET-207-6-0-0-1
Parent: NET-207-0-0-0-0
NetType: Direct Allocation
NameServer: PRI3.DNS.CA.TELUS.COM
NameServer: PRI4.DNS.CA.TELUS.COM
Comment:
RegDate:
Updated: 2002-04-08

TechHandle: PSINET-CA-ARIN

TechName: TELUS Communications Inc.
TechPhone: +1-613-780-2200
TechEmail: swip@swip.ca.telus.com

OrgAbuseHandle: AAT-ARIN
OrgAbuseName: Abuse at TELUS
OrgAbusePhone: +1-604-444-5791
OrgAbuseEmail: abuse@telus.com

OrgTechHandle: IA86-ARIN
OrgTechName: IP Admin, IP
OrgTechPhone: +1-403-503-3800
OrgTechEmail: add-req.tac@telus.com

OrgTechHandle: PSINET-CA-ARIN
OrgTechName: TELUS Communications Inc.
OrgTechPhone: +1-613-780-2200
OrgTechEmail: swip@swip.ca.telus.com

OrgTechHandle: TBOTP-ARIN
OrgTechName: TELUS BC ORG TECH POC
OrgTechPhone: +1-604-444-5791
OrgTechEmail: IPadmin@telus.com

Correlations for Optimum Online are provided by MyNetwatchman, which lists the ISP as high volume, garnering over 125 incidents per week. [OPTONLINE]. Port 137 was specifically listed as a target port for incident ids 15719213, 13051708, 16895880, 14828967, and 12197278 emanating from this ISP. As the MyNetwatchman service aggregates reports, each incident can represent hundreds of events. The correlation for the TELUS site was not nearly as strong.

The article by Bryce Alexander [ALEXANDER] provides data correlation for this event in his description of the dangers involved in allowing traffic to port 137. No external host should need to perform NETBIOS name lookups, so this would appear to be malicious in nature. While the University has heeded Beardsley's advice not to report internal port 137 traffic, it would appear that it is now being allowed from external sources, where it was not permitted at the time of Beardsley's analysis [BEARDSLEY]. The Dshield report for port 137 indicates that it is among the top reported ports during the same time period.

Further, there are several vulnerabilities reported in conjunction with port 137 by CVE [CVE], the most recent CVE-2001-1162. To quote:

> Directory traversal vulnerability in the %m macro in the smb.conf configuration file in Samba before 2.2.0a allows remote attackers to

overwrite certain files via a .. in a NETBIOS name, which is used as the name for a .log file.

Several other vulnerabilities refer to possible Denial of Service attacks.

Recommendation: Continue to monitor external accesses only with the Snort rule, but restore the block of port 137 traffic at the firewall.

Alert: Watchlist 000220 IL-ISDNNET-990517
Severity: Moderate
Frequency: 25,715

This is another custom Snort signature, not found in the default rule set.
A typical entry would be:

02/23-01:07:01.225344 [**] Watchlist 000220 IL-ISDNNET-990517 [**]
212.179.99.242:4682 -> MY.NET.240

Analysis of the traffic shows that it triggers on traffic from any host in the Class B 212.179 subnet. Various ports (more than 800) were accessed on the University network. The three most frequently appearing ports account for almost 50% of the traffic:

Port	Frequency	Percent	Cumulative%	Services
2561	6021	23.41	23.41	MosaixCC
6699	4039	15.71	39.12	Napster
1214	1826	7.10	46.22	Kazaa

The presence of the file sharing and music sharing programs, Napster and Kazaa (including Morpheus and Grokster) seem to indicate an abiding interest in sharing music, though it is surprising that so much of the activity comes from a single ISP. The predominant accesses, however, are to port 2561, associated with the Mosaix PDS (Predictive Dialing System). Now owned by Avaya, it appears to be an integrated call center management system combining telephony hardware and software that automates and synchronizes contact center activity. [AVAYA]. Searches of several Trojan port listings did not associate port 2561 with a known Trojan program nor was known vulnerability in Mosaix PDS itself found. The interest in this port is suggestive and further research is recommended. A wide variety of hosts from the same ISP seem to have an abiding interest in the University network.

This series of addresses was chosen for the third host registration lookup. The ISP appears to be located in Israel.

role: BEZEQINT HOSTMASTERS TEAM
address: bezeq-international
address: 40 hashacham
address: petach tikva 49170 Israel
phone: +972 1 800800110
fax-no: +972 3 9203033
e-mail: hostmaster@bezeqint.net
admin-c: YK76-RIPE
tech-c: MR916-RIPE
nic-hdl: BHT2-RIPE
remarks: Please Send Spam and Abuse ONLY to abuse@bezeqint.net
mnt-by: AS8551-MNT
changed: hostmaster@bezeqint.net 20021029
changed: hostmaster@bezeqint.net 20030204
source: RIPE

MyNetwatchman has more than 50 incidents listed in the history for this site.

Recommendation: Continue to monitor for hostile activity. If it has not already been done, make contact with the abuse handlers at the Israeli ISP and attempt to establish a collegial relationship. Research possible vulnerabilities associated with the Mosaix PDS and/or port 2561.

Alert: spp_http_decode - IIS Unicode attack detected
Severity: Noise
Frequency: 13,373
Snort Signature: http_decode

While there are about 30 different practicals on the GIAC site that reference this Snort signature, it is not explicitly listed as a separate signature on the Snort site. The Snort FAQ explains that the messages are produced by the http_decode preprocessor and that normal surfing with Netscape can trigger them. [SNORT] The apparent reason for this is that Netscape includes Unicode characters in its cookies. [GELMAN] Filtering outbound alerts only is recommended as a work around.

The Snort Users Manual indicates that the http_decode pre-processor processes HTTP URL strings and converts them to ASCII strings. It enables detection of attacks associated with various Unicode translation tricks for directory traversal. Code Red and Nimda are specific examples of such exploits. As readers will doubtless remember, Nimda spewed huge amounts of network traffic, amount to a large scale denial of service attack. The original Code Red worm defaced web pages and sought other machines to infect [MCAFEE].

An example of an alert is included next for the sake of illustration.

```
02/25-00:15:56.160149  [**] spp_http_decode: IIS Unicode
attack detected [**] MY.NET.224.166:4633 ->
216.35.123.105:80
```

Of the alerts recorded, only 662 represented inbound traffic and thus may have been harmful to the University net. Almost 13,000 were outbound and constitute noise. Normal web browsing can trigger the alert, especially using the Netscape browser. One host, MY.NET.207.24, is responsible for 1,712 of the alerts. Using a vulnerability scanner would be a better means of determining if a host is infected.

Recommendation: As per the recommendations on the Snort web page, filter outbound traffic so that only inbound alerts are recorded. Use a vulnerability scanner to pinpoint problem hosts. Follow up on MY.NET.207.24 for possible problems and remediation.

Alert: High port 65535 tcp - possible Red Worm - traffic
Severity: High
Frequency: 11,479

This was another signature that appears to be non-standard as it is not in the current set of Snort signatures. It may be another custom signature. Here is a sample alert:

02/25-00:16:34.049953 [**] High port 65535 tcp - possible Red Worm - traffic [**]
62.212.112.98:63100 -> MY.NET.88.193:65535

The attacker is seeking a host with a backdoor already installed and bound to port 65535. In that case, root access would be allowed to anyone connecting to port 65535 via telnet. The Adore worm attempts to gain unauthorized access to systems that are vulnerable to the LPRng, rpc-statd, and the Berkeley Internet Name Domain (BIND) software exploits. Once the Adore worm has gained access to a system, introduces a Trojan called “icmp” opens a backdoor on TCP port 65535 to the system when a specific packet is received. If the worm successfully installs itself, it emails critical information about the infected system (to either adore9000@21cn.com and adore9000@sina.com, or adore9001@21cn.com and adore9001@sina.com). This worm also randomly generates the a class B subnet and then scans that entire subnet for any other vulnerable systems, seeking to propagate further [DELL]. Half of the roughly 12,000 alerts are internal, and half are external. However, the internal addresses represent only 110 unique hosts in the University network. This would lead to the conclusion that these hosts are infected and actively scanning.

The port does not show up on either Dshield or MyNetwatchman as current active target. As the exploit is somewhat dated, it is surprising to find an active outbreak. Either the machines were never actually cleaned up, or this is some

new vulnerability with enough similar charactistics to take advantage of the old vulnerabilities.

Recommendations: Use a vulnerability scanner to seek machines with port 65535 listening and responsive to telnet, investigate for possible Adore infection and secure those hosts. Disable incoming telnet in favor of Secure Shell (and other secure services) across the entire University.

Using a one line shell script to isolate the most frequent source host in the University gives the result MY.NET.88.193. A one line awk script was used to isolate events where this host was the source [AHO] and the output was read into SAS for frequency analysis.

```
cat all.data | awk –F% '$4 ~ /MY.NET.88.193/ {print $0}' | cut –d% -f6
```

Sixteen targets were identified, all of them outside of the University network. The top three targets are also communicating back with MY.NET.88.193 on the same 65535 port. This is illustrated in the graphic on the next page.

Frequent Scans

UDP Scan from 130.85.218.62
Severity: Noise
Snort Signature: spp_portscan
Frequency: 29,359 events

The most ubiquitous scans come from 130.85.218.62. The scan starts at 03:30 and ends at 20:04 on February 26, averaging 29 events per minute over the course of the day. The destinations are outside the University, and the vast majority of the traffic is UDP. A UDP scan will send a UDP packet to each port on a target. An open UDP port will accept the packet and sending no reply, while a closed port responds with an ICMP unreachable packet [MILLICAN]. In short, it is an attempt to map parts of another subnet via a UDP scan, and it is quite a noisy one. A short but typical excerpt from the almost 30,000 events is below:

```
Feb 26 05:51:35 130.85.218.62:1090 -> 80.232.11.236:8467 UDP
Feb 26 05:51:35 130.85.218.62:1126 -> 80.232.11.236:24322 UDP
Feb 26 05:51:35 130.85.218.62:1224 -> 80.232.11.236:26978 UDP
Feb 26 05:51:35 130.85.218.62:1235 -> 80.232.11.236:2616 UDP
Feb 26 05:51:35 130.85.218.62:1238 -> 80.232.11.236:12126 UDP
```

Any number of scanning tools could have been used by the person responsible for the host that originated the scan. Since originating ports increment by different values rather than linearly increasing, it can be inferred that the scan is targeting hosts outside of the university as well as those within.

A lookup on this host shows that the University providing the data is part of the University of Maryland system:

OrgName: University of Maryland Baltimore County
OrgID: UMBC
Address: UMBC University Computing
City: Baltimore
StateProv: MD
PostalCode: 21250
Country: US

The hostname resolves to resnet2-89.resnet.umbc.edu. This would appear to be a machine in a student dorm room. Most likely it is a violation of the Acceptable Use Policy. Since the scan was noisy, it might be reported to the University abuse handlers, but more likely after the person attempts to take advantage of the results of the reconnaissance.

Recommendations: The Acceptable Use Policy should be clarified for the person responsible for the scanning activity. Records of possible past infractions should be checked. General user awareness training for students on the Acceptable Use Policy of the University would be advisable as well.

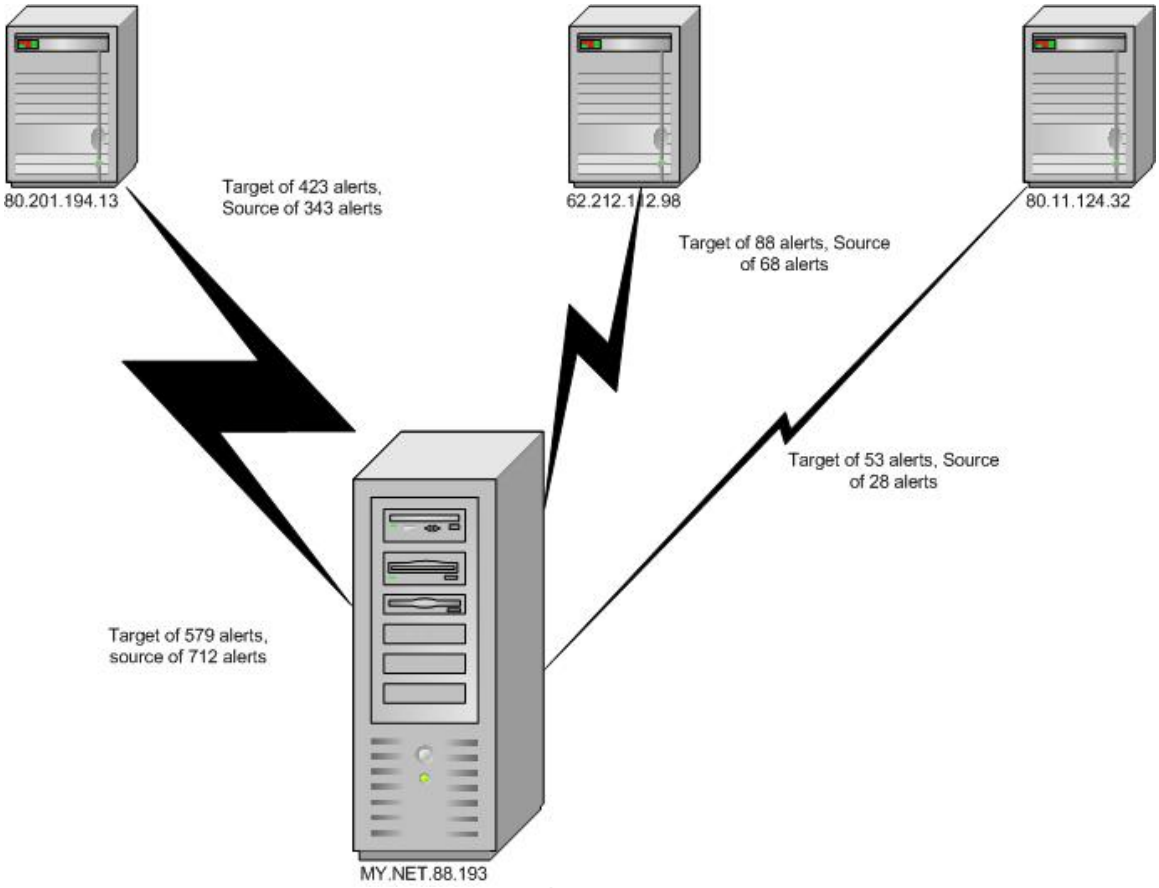


Figure: Communication rates between MY.NET.88.193 and top targets on port 65535

Out of Spec Packets

While the "out of spec" packets for the 5 day period make up the smallest of the three sets of files, it weighs in at almost 10,000 lines (alerts and scans are an order of magnitude higher). This network experiences between 10% and 15% malformed packets. The vast majority, 11,325 are from outside of the University, with only 141 from inside. The first line of each entry was isolated with

grep "\->" oall.data

The traffic from the outside is not under University control, so internal traffic will be the focus here. Using awk to isolate those internal to the network:

grep "\->" oall.data | awk ' $2 ~ /MY.NET/ { print $0 }'

we find that 75 are generated by MY.NET.208.230, mostly on port 21, the port for File Transfer Protocol (FTP) [IANA]. They start at 21:13 on February 25 and end at 02:12 the next day, spanning a period of almost exactly 5 hours. The target is 217.9.113.66; this provides a good candidate for an additional host registration lookup. The ISP is evidently in Germany.

role: Completel Hostmaster
address: CompleTel GmbH
address: Hans-Stiessberger-Strasse 2b
address: D-85540 Haar b. Muenchen
address: Germany
phone: +49 89 95465 0
fax-no: +49 89 95465 889
e-mail: hostmaster@completel.de
admin-c: MH16594-RIPE
tech-c: PB583-RIPE
tech-c: DW193-RIPE
tech-c: BS1369-RIPE
tech-c: TS1544-RIPE
nic-hdl: CH6086-RIPE
mnt-by: COMPLETEL-RIPE-MNT
changed: hostmaster@alphacom.de 20000705
changed: fritz.reichmann@completel.de 20020108
changed: randy@ipcenta.de 20020409
changed: Michael.Ferwagner@CompleTel.de 20020827
changed: Bernhard.Schmidt@completel.de 20030205
source: RIPE

person: Milislav Radmanic
address: Deutschherrnstr. 15- 19
address: D-90429 Nuernberg
address: Germany
phone: +49 911 74053 0
e-mail: radmanic@suse.de
nic-hdl: MR1262-RIPE
mnt-by: COMPLETEL-RIPE-MNT
notify: hostmaster@completel.de
changed: Michael.Ferwagner@CompleTel.de 20020419
source: RIPE

Each of the 75 records carry the same 5 TCP options:

```
TCP Options (5) => MSS: 1460 SackOK TS: 23737 0 NOP WS: 0
```

The options, translated [SANS]:

1. Maximum Segment Size (MSS) is set to 1460.
2. Selective Acknowledgement (SackOK) is permitted
3. Window Scale (WS) is 0
4. Time Stamp (TS) is recorded
5. No operation (NOP) pads out the data to the byte boundary

The value of MSS is maximum length of the TCP portion of the segment. The SackOK permits retransmission of only missing datagrams, rather than forcing the retransmission of all datagrams following the missing one. [MATHIS] WS is sent only in a SYN segment and is negotiated between sender and receiver as part of the initial 3 way handshake; the values must agree for communication to take place. [JACOBSON] Timestamp (TS) implements Protect Against Wrapped Sequence Numbers (PAWS) by avoiding old duplicate datagrams from the same connection. Timestamp also enables Round Trip Time Measurement (RTTM), designed to improve reliability and decrease unnecessary retransmission.

The port numbers on MY.NET.208.230 steadily increment from 64687 to 65060, drop back to 61025 and begin incrementing again. So, the traffic would appear to be a port scan or reconnaissance of some sort. The technique is described in the classic paper by Fyodor on fingerprinting. [FYODOR] Fyodor details the creative use of the timestamp option as follows:

> Another number that can be sequenced for OS detection purposes is the TCP timestamp option values. Some systems do not support the feature, others increment the value at frequencies of 2HZ, 100HZ, or 1000HZ, and still others return 0. Nmap also uses this information to determine uptime values for the remote host.

Further down he notes that Linux tends to return the same MSS value that is initially sent.

Checking the alerts file for MY.NET.208.230 yields over 30 signatures, including 8 detected portscans and 4 IIS Unicode attacks. Duplicated attacks have been removed so that the resulting lines are more illustrative of the traffic that has been logged.

```
02/24-23:01:59.208215  [**] spp_portscan: PORTSCAN DETECTED from
MY.NET.208.230 (STEALTH) [**]
```

```
02/24-23:02:02.477802  [**] spp_portscan: portscan status from
MY.NET.208.230: 1 connections across 1 hosts: TCP(1), UDP(0) STEALTH
[**]
02/24-23:02:10.215157  [**] spp_portscan: End of portscan from
MY.NET.208.230: TOTAL time(0s) hosts(1) TCP(1) UDP(0) STEALTH [**]
02/25-00:17:40.486906  [**] EXPLOIT x86 NOOP [**] 217.9.113.66:22875 ->
MY.NET.208.230:61242
02/25-00:21:07.468829  [**] SMB Name Wildcard [**] 211.8.4.180:1026 ->
MY.NET.208.230:137
02/25-10:03:15.051491  [**] SMB Name Wildcard [**] 200.185.91.156:1026
-> MY.NET.208.230:137
02/26-06:35:36.271689  [**] spp_http_decode: IIS Unicode attack
detected [**] 24.188.217.73:2325 -> MY.NET.208.230:80
02/26-06:35:36.271689  [**] spp_http_decode: IIS Unicode attack
detected [**] 24.188.217.73:2325 -> MY.NET.208.230:80
02/26-06:35:36.271689  [**] spp_http_decode: IIS Unicode attack
detected [**] 24.188.217.73:2325 -> MY.NET.208.230:80
02/26-06:35:36.271689  [**] spp_http_decode: IIS Unicode attack
detected [**] 24.188.217.73:2325 -> MY.NET.208.230:80
02/26-19:50:57.354190  [**] SMB Name Wildcard [**] 200.67.231.168:1026
-> MY.NET.208.230:137
```

The host MY.NET.208.230 is decidedly suspicious. It has almost certainly been hacked. A security audit is in order for the host as it may have been compromised and is now being used for further reconnaissance. At the least, the port scanning traffic is indicative of possible violations of the Acceptable Use Policy.

Top Ten Talkers

These were chosen as the top ten talkers as they were responsible for the most alerts among them. It is significant that 5 addresses from the Bezeqint ISP referenced previously are among the top talkers. This would explain how the ISP became the subject of a custom alert.

Address	Frequency	Percent
212.179.94.48	6021	3.95
212.179.13.98	3502	2.30
202.175.95.50	2545	1.67
67.81.224.77	2007	1.32
MY.NET.207.34	1712	1.12
212.179.102.22	1574	1.03
212.179.100.234	1414	0.93
212.179.35.118	1252	0.82
MY.NET.241.182	1151	0.76
MY.NET.244.78	1099	0.72

Additional alerts

Alert	Frequency	Percent
CS WEBSERVER - external web traffic	9742	6.39
Port 55850 tcp - Possible myserver activity - ref. 010313-1	4996	3.28
spp_http_decode - CGI Null Byte attack detected	2459	1.61
Tiny Fragments - Possible Hostile Activity	1748	----
SUNRPC highport access!	1663	1.09
TFTP - Internal TCP connection to external tftp server	1538	1.01
TFTP - External UDP connection to internal tftp server	1450	0.95

The next 6 alerts in order account for little in terms of the overall traffic. All appear to be custom alerts except for the CGI Null Byte attack, associated with the preprocessor (see the detect on IIS Unicode above). Without further details as to the specific rules, it is difficult to make more than general statements about this group of alerts.

The CS WEBSERVER alert appears to trigger on port 80 accesses to MY.NET.100.165. To date, almost 20 other analysts noted the presence of this alert, so it has been in place for some time. A sample alert follows for purposes of illustration:

```
02/23-00:45:38.217423  [**] CS WEBSERVER - external web traffic [**]
202.175.95.50:44036 -> MY.NET.100.165:80
```

Dan Hawrylkiw notes it is a watch to monitor external access to the "CS webserver" and further recommended removal of the alert [HAWEYLKIW]. If the alert is in place simply to keep a record of accesses to the web server, the logging facilities of the web server itself are better suited to the task.

To look at traffic patterns as a whole, the following script was used to derive the alerts that were targeted to port 80.

```
cat all.data | awk –F% '$7 ~ /80/ {print $0}'
```

This selected 27,315 records out of the total of 152,414, or roughly 20% of the traffic. The data was imported to SAS and a frequency analysis was run. The results showed the host 202.175.95.50 to be the most prolific. However, this host does not appear in the scan or out of spec files. Port 4888 seems to be the most common source port, appearing 432 times, all but one from 212.179.83.24, one of the hosts in the Israeli ISP. Port 4888 does not appear to be associated with a particular known service or port, but would bear some further investigation.

The next alert for Port 55850 is also a custom signature. Analyst Chistof Voemel [VOEMEL] pointed out that there is a MyServer DDOS agent listening on UDP port 55850 as noted in http://archives.neohapsis.com/archives/incidents/2000-10/0136.html (the posting is still available). However, as Voemel points out, this posting centers on UDP traffic, and the actual alert appears to be associated with the TCP protocol. Looking at the traffic broadly, the rule alerts on either incoming or outgoing traffic for port 55850. Ports 1214 (Kazaa), 3724 (Blizzard Battlenet) and 5190 (America-Online) are paired frequently (about 300 times each) with the port 55850 traffic, although not clearly neither exclusively nor predominantly. No strong patterns in source or destination addresses emerge, either. Other GCIA analysts have been puzzled as well. The network analyst at the target site may be able to provide some further insight.

Another false positive is found in the "spp_http_decode - CGI Null Byte attack detected". This is not a signature per se, but is part of the http preprocessor code. Basically, if the http decoding routine finds a %00 in an http request, it will alert with this message. [SNORT].

Examining the alerts data indicates that 2,420 of the 2,460 alerts are generated internally, from MY.NET addresses.

```
grep "CGI Null" alert*.out |  wc –l
grep "CGI Null" alert*.out | cut –d% -f3 | grep MY.NET | wc –l
```

The Snort FAQ [SNORT] indicates that internal users can trigger these, and Netscape in particular is prone to do so. The documentation goes on to recommend that outbound port 80 traffic be filtered so that only external hits from this signature are logged. This will allow analysts to focus on what are more likely actual attacks.

The "Tiny Fragments" alert was inserted here as it would have ranked seventh overall in terms of count. Again, this appears to be a As noted above, these alerts did not contain a port number and so were removed in order to normalize the data for statistical analysis.

```
grep Tiny alert*.out | grep MY.NET.246.54 | wc –l
```

Of the 1,748 hits, 1511 are generated from the internal host MY.NET.246.54. Since the description flags the alert as "Possible Hostile Activity" and it seems to come from within, the host MY.NET.246.54 would likely benefit from a security review. As to correlation, the alert appears in the work only three other analysts, those being Mike Bell, David Singer and Andy Siske. From that, a recent uptick in this activity is surmised, although apparently the alert has been in place for some time. It may be some sort of reconnaissance aimed at an attempt to bypass firewalls or intrusion detection systems. [SISKE]

Additionally, MY.NET.246.54 added 381 PORTSCAN DETECTED alerts to the alerts file and 4 SMB Name Wildcard alerts (as discussed elsewhere). This activity is further reflected in the scans files with 1416 alerts noted. As the dotted quad notation is used in the scans file instead of the MY.NET.246.54 notation, this relationship was not immediately noted. A security audit is recommended for this host as soon as possible.

SUNRPC is associated with the CAN-2002-0391 integer overflow [CVE] and remote attackers could execute arbitrary code, so this custom alert bears further investigation. All of the hits are on port 32771 on various hosts in the internal network. The range of ports for RPC services is 32770-32900. It is interesting to note that 712 of the 1665 alerts, or about 42%, originated from port 2101. This port is associated with Digital Global Positioning Systems. [PORT, WSRCC]. Although it does not appear to be malicious, it is interesting behavior to note for further investigation.

The set of two TFTP alerts are important indicators of possible problems as well. The Trivial File Transfer Protocol (TFTP) should not cross the network perimeter. It is used for file transfer but there is no user authentication. [SOLLINS] If a host provides tftp without restricting the access, an attacker can read and write files anywhere on the system. The example shows retrieval of a remote password file but other files (and Trojans) could be retrieved or deposited using the same technique.

```
 evil % tftp
tftp> connect victim.com
tftp> get /etc/passwd /tmp/passwd.victim
tftp> quit
```

This service is one that should be turned off unless it is specifically needed. If it is necessary, access should be restricted to a directory that has no valuable information. [FARMER]. More recently, tftp was identified as one of the six attack vectors for the Nimda Worm. Other attack vectors are listed as:

- E-mail attachments
- IE browsing an infected IIS Web server with JavaScript enabled
- Sharing the C:\ drive of an infected system
- Web folder traversal vulnerability on IIS servers
- Highlighting a file with extension .eml or .nws in Windows Explorer with Active Desktop enabled. [COUNTERPANE]

The site would be well advised to block incoming and outgoing tftp service (69/udp) at the firewall. [IANA].

A high level of network craftsmanship is demonstrated by writing custom Snort rules. However, the significance of the rules is not obvious to the outside

observer. It would be an excellent audit practice to review the necessity of each of these rules to determine their continued usefulness.

Conclusions and Recommendations

In conclusion, the site seems to be in better shape than it was about a year ago in May when Beardsley did his analysis [BEARDSLEY]. The administrators appear to have heeded much of his advice with regard to muffling noisy alerts. However, opening port 137 on the University firewall is evidently a mistake, and this should be addressed promptly. Other services such as TFTP (port 69) should be blocked as well.

The CS WEBSERVER signature generates a significant number of extra alerts Better web access monitoring can be provided by web server logs so this signature should be dropped. Filtering should also be put in place for http traffic to minimize false positives on the CGI Null Byte attack. The activity on port 65535 should be followed up with targeted vulnerability scanning to eliminate it from the internal network. Infected hosts should be cleaned and secured. A review of the Snort rule set, especially the custom rules, should be undertaken on a regular basis. The significant number of alerts from the 212.179 network in Israel should continue to be monitored and efforts made to work with the ISP towards resolution of the continuing attacks. Further research into vulnerabilities involving port 2561 and/or the Mosaix Predictive Dialing System are recommended as well as port 4888, used frequently in attacks on port 80.

There are several recommendations that apply specifically to hosts. Those hosts that can be identified as infected or problematic from the alerts should be followed up promptly. Specifically, the host MY.NET.246.54 triggers the custom alert “Tiny Fragments – Possible Hostile Activity” and should undergo a security audit to remedy any issues. It also appears in the scan files. The host MY.NET.207.24 is responsible for a significant number of alerts on the IIS Unicode attack and should also be audited. Finally, the host MY.NET.208.230 is responsible for scanning activity. Minimally, review the Acceptable Use Policy with the person responsible for the host and evaluate whether or not an audit is needed.

Introduction of secure services (ssh, sftp, and so forth) would significantly improve the security posture. Any unnecessary services, such as TFTP, should be weeded out and disabled.

User awareness is the first line of defense for the contemporary network. Education on computing policies and appropriate Internet usage is a perennial good investment for the future.

References

- [AHO] Aho, Alfred V., Kernighah, Brian W., Weinberger, Peter J. The AWK Programming Language Reading: Addison-Wesley Publishing Company. 1988.

- [ALEXANDER] Alexander, Bryce. "Port 137 Scan." Intrusion Detection FAQ. May 10, 2000. http://www.sans.org/newlook/resources/IDFAQ/port_137.htm (March 8, 2003).

- [AVAYA] "A Key to Keeping Your Customers: Integrated Call Logging" http://www1.avaya.com/enterprise/whitepapers/gcc1038.pdf. 2000. (May 11, 2003)

- [BEARDSLEY] Beardsley, Tod A. "Intrusion Detection and Analysis: Theory, Techniques, and Tools" GIAC Certified Intrusion Analysts (GCIA). http://www.giac.org/practical/Tod_Beardsley_GCIA.doc (March 8, 2003).

- [COUNTERPANE] "Nimda Worm" Counterpane Security Alerts http://www.counterpane.com/alert-nimda.html September 18, 2001 (May 11, 2003)

- [CVE] Common Vulnerabilities and Exposures http://www.cve.mitre.org/ (March 8, 2003).

- [DELL] Dell, J. Anthony "Adore Worm – Another Mutation" http://www.sans.org/rr/threats/mutation.php April 6, 2001 (March 8, 2003).

- [DSHIELD] Dshield Port Report http://dshield.org/port_report.php?port=137 (March 8, 2003).

- [FARMER] Farmer, Dan and Venema, Wietse "Improving the Security of Your Site by Breaking Into It" http://www.porcupine.org/satan/demo/docs/admin_guide_to_cracking.html (May 11, 2003).

- [FYODOR] Fyodor "Remote OS detection via TCP/IP Stack Fingerprinting" October 18, 1998 http://www.insecure.org/nmap/nmap-fingerprinting-article.html (March 8, 2003).

- [GEEKTOOLS] Geektools http://geektools.com (March 8, 2003)

- [GELMAN] "Flooded with IIS Unicode and CGI Null Byte false positives" Snort-users mailing list. http://archives.neohapsis.com/archives/snort/2001-02/0093.html Neohapsis Archives. February 6, 2001. (May 11, 2003).

- [HAWRYLKIW] Hawrylkiw, Dan "Intrusion Detection in Depth: GCIA Practical Assignemtn Version 3.0" GIAC Certified Intrusion Analysts (GCIA). March 25, 2003 http://www.giac.org/practical/Dan_Hawrylkiw_GCIA.doc

- [MATHIS] Mathis, M., Mahdavi, J., Floyd, S., Romanow, A. "RFC2018: TCP Selective Acknowledgement Options". October, 1996. ftp://ftp.rfc-editor.org/in-notes/rfc2018.txt. March 8, 2003.

- [NEWPORT] Newport, Brandon "Level Two Intrusion Detection In Depth GCIA Practical Assignment Version 2.7a" GIAC Certified Intrusion Analysts (GCIA). March 8, 2003 http://www.giac.org/practical/Brandon_Newport_GCIA.zip

- [IANA] Port Numbers http://www.iana.org/assignments/port-numbers (March 8, 2003).

- [JACOBSON] Jacobson, V., Braden, R., Borman, D. "RFC1323: TCP Extensions for High Performance" May 1992 ftp://ftp.rfc-editor.org/in-notes/rfc1323.txt (March 8, 2003).

- [MCAFEE] Virus Information Library McAfee http://vil.nai.com/ (March 8, 2003).

- [MICROSOFT 1] "MS02-045: Unchecked Buffer in Network Share Provider May Lead to Denial-of-Service" http://support.microsoft.com/default.aspx?scid=kb;en-us;326830 Last reviewed: 4/22/2003. (May 11, 2003)

- [MICROSOFT 2] "CIFS: A Common Internet File System" http://www.microsoft.com/mind/1196/cifs.asp November, 1996. (May 11, 2003)

- [MILLICAN] Millican, Andy "Network Reconnaissance – Detection and Prevention" GIAC Security Essentials Certification http://www.giac.org/practical/GSEC/Andy_Millican_GSEC.pdf January 23, 2003 (March 8, 2003).

- [OPTONLINE] ISP Ratings (optonline.net) http://www.mynetwatchman.com/LIS.asp?Queue=HBRD&PID=201 (March 8, 2003).

- [PORT] Port Lookup Utility http://www.treachery.net/security_tools/ports/ (March 8, 2003).

- [ROESCH] Roesch, Martin and Green, Chris Snort Users Manual, Snort Release 1.9.1 http://www.snort.org/docs/SnortUsersManual.pdf November 11, 2002 (March 8, 2003).

- [SAMSPADE] Sam Spade http://www.samspade.org (March 8, 2003).

- [SANS] "TCP/IP and tcpdump Pocket Reference Guide" Bethesda: SANS Institute. 2002.

- [SAS] SAS Institute http://www.sas.com (March 8, 2003).

- [SISKE] Siske, Andrew "GIAC Intrusion Detection Practical Assignment for SANS Security DC 2000" GIAC Certified Intrusion Analysts (GCIA). March 8, 2003 http://www.giac.org/practical/Andy_Siske_GCIA.htm

- [SNORT] Snort http://www.snort.org/ (March 8, 2003).

- [SOLLINS] Sollins, K. "RFC 1350: The TFTP Protocol (Revision 2)". July 1992. ftp://ftp.rfc-editor.org/in-notes/rfc1350.txt (March 9, 2003).

- [VOEMEL] Voemel, Christof "SANS Intrusion Detection Practical – SANS Parliament Square 2001" GIAC Certified Intrusion Analysts (GCIA). March 8, 2003 http://giac.org/practical/Christof_Voemel_GCIA.txt

- [WSRCC] WSRCC March 8, 2003 http://www.wsrcc.com/wolfgang/gps

Last Updated: May 29th, 2017

Upcoming Training

Security Operations Center Summit & Training	**Washington, DC**	**Jun 05, 2017 - Jun 12, 2017**	**Live Event**
SANS Houston 2017	**Houston, TX**	**Jun 05, 2017 - Jun 10, 2017**	**Live Event**
SANS Columbia, MD 2017	**Columbia, MD**	**Jun 26, 2017 - Jul 01, 2017**	**Live Event**
SANS London July 2017	**London, United Kingdom**	**Jul 03, 2017 - Jul 08, 2017**	**Live Event**
SANSFIRE 2017	**Washington, DC**	**Jul 22, 2017 - Jul 29, 2017**	**Live Event**
SANSFIRE 2017 - SEC503: Intrusion Detection In-Depth	**Washington, DC**	**Jul 24, 2017 - Jul 29, 2017**	**vLive**
SANS San Antonio 2017	**San Antonio, TX**	**Aug 06, 2017 - Aug 11, 2017**	**Live Event**
SANS Boston 2017	**Boston, MA**	**Aug 07, 2017 - Aug 12, 2017**	**Live Event**
SANS Adelaide 2017	**Adelaide, Australia**	**Aug 21, 2017 - Aug 26, 2017**	**Live Event**
SANS Virginia Beach 2017	**Virginia Beach, VA**	**Aug 21, 2017 - Sep 01, 2017**	**Live Event**
SANS Network Security 2017	**Las Vegas, NV**	**Sep 10, 2017 - Sep 17, 2017**	**Live Event**
SANS vLive - SEC503: Intrusion Detection In-Depth	**SEC503 - 201709,**	**Sep 11, 2017 - Oct 18, 2017**	**vLive**
SANS Baltimore Fall 2017	**Baltimore, MD**	**Sep 25, 2017 - Sep 30, 2017**	**Live Event**
SANS London September 2017	**London, United Kingdom**	**Sep 25, 2017 - Sep 30, 2017**	**Live Event**
Baltimore September 2017 - SEC503: Intrusion Detection In-Depth	**Baltimore, MD**	**Sep 25, 2017 - Sep 30, 2017**	**vLive**
Community SANS Scottsdale SEC503	**Scottsdale, AZ**	**Oct 02, 2017 - Oct 07, 2017**	**Community SANS**
SANS October Singapore 2017	**Singapore, Singapore**	**Oct 09, 2017 - Oct 28, 2017**	**Live Event**
Community SANS Boston SEC503	**Boston, MA**	**Oct 09, 2017 - Oct 14, 2017**	**Community SANS**
Community SANS Ottawa SEC503	**Ottawa, ON**	**Oct 16, 2017 - Oct 21, 2017**	**Community SANS**
SANS San Diego 2017	**San Diego, CA**	**Oct 30, 2017 - Nov 04, 2017**	**Live Event**
San Diego Fall 2017 - SEC503: Intrusion Detection In-Depth	**San Diego, CA**	**Oct 30, 2017 - Nov 04, 2017**	**vLive**
SANS Seattle 2017	**Seattle, WA**	**Oct 30, 2017 - Nov 04, 2017**	**Live Event**
SIEM & Tactical Analytics Summit & Training	**Scottsdale, AZ**	**Nov 28, 2017 - Dec 05, 2017**	**Live Event**
SANS Cyber Defense Initiative 2017	**Washington, DC**	**Dec 12, 2017 - Dec 19, 2017**	**Live Event**
SANS OnDemand	**Online**	**Anytime**	**Self Paced**
SANS SelfStudy	**Books & MP3s Only**	**Anytime**	**Self Paced**