Global Information Assurance Certification Paper

GIAC Intrusion Detection In Depth
Practical Assignment
Version 3.0

Assignment 1: State of Intrusion Detection

Comparison of Host-Based to Network Based IDS

Randall L. Gillespie

February 25, 2002

Table of Contents

Overview of Intrusion Detection Systems

In most cases, there are two types of intrusion detection systems, host-based and network-based. When comparing the two types of systems there are many similarities and differences. In addition, when used together they compliment each other and can fill the void where the other cannot effectively accomplish its intended objectives. This practical will show the benefits, challenges, similarities, and differences that exist between host-based and network-based ID systems.

Host-Based Intrusion Detection Systems

Host-based systems tend to seek out attack signatures in log files, as well as require a client-loaded piece of software on the system in order to be monitored. Host-based IDS involves not only monitoring the network or internet traffic into and out of a single computer, but can also involve the checking of the integrity of system to include files and monitoring for suspicious processes.

There are two leading classes of host-based intrusion detection software: host wrappers/personal firewalls, and agent-based software. Both of these approaches are effective in detecting trusted-insider attacks, which can be missed by network-based ids systems.

Host-wrappers/firewalls can be configured to look at all the network packets, connection attempts, or login attempts to a monitored system. This type of ID method in some cases of software can be used to thwart active attacks. Tools such as TCPwrappers are able to allow or reject specific hosts/internet addresses from accessing the services on the system. Personal firewalls also give this same ability including the ability to detect the piece software that is making the connection to the network. Both of these individual solutions do not normally provide a "corporate" solution. These solutions tend to be better suited for bastion hosts and if combined with a network-based IDS platform can provide a very good start on your network intrusion detection and security platform.

Some host-based agents are capable of monitoring system conditions, to include changes to system files, changes in user privileges, and various other system changes. These systems alert the administrator to the changes via an alerting mechanism such as administrative pages or other communication method. An example of one of the tools would be tripwire. (**http://www.tripwiresecurity.com**). Most flavors of Unix for example have built in tools such as netstat, lsof, top, and others that can allow the system administrator to know what is going on with their system. These tools can allow the administrator to get to know the system to determine what is normal or abnormal, and to establishing a baseline of normal activity. When the baseline is established abnormal activity can be more easily identified, and investigated.

Benefits of Host-Based Intrusion Detection Systems

Host-based intrusion detection systems are able to provide significant benefits to your security infrastructure. A major benefit provided by host-based intrusion detection is that it will add a targeted layer of security to vulnerable systems. In some cases, the host-based ID system may be able to halt the attack from continuing against the "protected" system. In addition to the added layer of security, is its ability to detect both internal as well as attacks originating from external sources. This feature is not found in most

firewalls and/or network monitors, or they do not have the capability to perform this for end hosts. The same originating location of the attack makes a host-based solution an especially attractive feature due to the fact a majority of the attacks you will encounter will arise from an internal source rather than an external hacker.

An added benefit of host-based intrusion detection is its proximity to authenticated users. This close proximity to the user allows for excellent trend detection as well as more detailed damage assessment. In addition, encryption does not normally pose a problem since the traffic is already decrypted by the host system, and the keys are present on the system.

Host-based ID systems are able to identify, track, and address attacks directed at your host concerning confidentiality, integrity, and availability.

Confidentiality

In the area of security problems concerning confidentiality, the host-based ID systems are able to detect unauthorized access to files, violations in corporate system use policies, violation of corporate security policies, as well as weak or non-existent passwords. An example of this would be an abuse of privilege attack, elevation of privileges, old account access, "backdoor" accounts placed by administrator, critical data access and modification or theft, changes in security configuration, and more. As noted a majority of these problems can be attributed to specific networks and would not be easily identified by a network intrusion detection system. In addition, these situations pose critical security risks for your network as well as possible legal ramifications.

Abuse of Privilege attack:

This type of attack falls under the category where a user already has the administrative capability and privileges. This user would than use those privileges in a manner that is in violation of corporate, security policies, or other unethical/unauthorized manner. An example of this could be a Microsoft Exchange Administrator using his/her privileges to access other mailboxes of executives or Human resources personnel. This constitutes a critical violation and very easily can result in legal ramifications. A host-based ID has the capability to identify this type of threat where it would be nearly impossible for a network intrusion detection system to identify this type of threat to the infrastructure.

Elevation of Privileges:

A host-based ID system possesses the capability to detect when a users privilege has been escalated by an administrator, either inadvertently, or via other means such as an attack. An example of this would be: A user runs some malicious code and raises their permission level. This type of action can be quickly identified, resolved, and dealt with if the host that is affected has host-based IDS software installed.

Old Account Access

Access of old accounts that are still active are especially common during times of recession or economic downturn, when there are many RIFs (reduction in force). Most organizations have policies and procedures to deal with the deletion and disabling of

accounts, when a person exits. Unfortunately, these procedures take time to make their way through the system and reach the appropriate administrator or group responsible for dealing with this. This leaves a window of opportunity for the user to log in and access corporate information. These accounts are open windows begging someone to enter and make off with crucial data of your organization and/or deletion/destruction of data. With host-based IDS in place, this type of activity is quickly identified and resolved. An example of this is accounts that are "flagged". When the account logs in, they cause an event to be generated from the IDS system. This can notify the appropriate person to investigate the access.

Backdoor Accounts/Undocumented/Unknown Accounts

In most organizations, there are procedures and policies in place regarding Administrator or root accounts. Sometimes administrators will install software that requires a privileged account and the administrator will create the account without considering the full ramifications of their action. Since the system administrator or software, being installed is installed while logged on as the administrator the ability to create additional accounts with these permissions, without going through the appropriate procedures, is present. This presents a valid risk since the newly created account is undocumented and only known by the system administrator. For example take the situation of a reduction in force (RIF) or the Administrator becomes upset and departs the organization. There is no record of the account creation which poses a real and valid risk much the same as "Old Account Access" where the window is open but since the account is unknown there is no way to account for this. A host-based ID system can assist in locating accounts of this type. Another common source of backdoor accounts are built into software. All software that is being installed should be verified and checked to ensure that existing policies and procedures are met, and the special accounts required are secured and noted.

Critical Data Access, Modification or Theft

In every business, there is data that is critical to the operations of the business. This data is commonly identified as mission-critical data. Some examples of this would be the website, databases, email, proposal information, and other types of information that is needed for the business to operate. This highly sensitive and/or classified information is deemed mission-critical, and if modification or release of this information occurred, it would pose critical security risks to the business, and possibly present significant liability. Imagine if someone released your human resources records to the public or everyone in the company had access to this information. For instance, if a disgruntled employee decided to modify certain information would this be considered critical? These actions could be very detrimental to a company. With a host-based ID system in place, this activity could be quickly identified and possible damage control measures implemented promptly.

Legal Issues

There is a legitimate need for legal notices on all network points of entry into the network. Usually simple items like this are missed. These small items can cause severe problems if it comes down to prosecution of an attacker. A defense attorney will analyze

every step you have taken and attempt to prove that if you are not able to accomplish small simple tasks how can you even begin to prove that the complex task that was perpetrated was actually done by his client.

Integrity

Some versions of a host-based ID system are able to check the integrity of system files. What this means is that the system will check the files via a checksum gathered when the agent was installed on the system, and than routinely check the system to see if changes were made and if their were changes they will check that against attack signatures to determine if malicious software has been installed. This if applied properly allows the integrity of the system files to be verified and monitored against possible malicious tampering of system and critical files. In addition, the integrity of the system itself can be verified to ensure there are no Trojans or malicious software arriving onto the system. If potential mal-ware is detected, the prompt notification can assists in the troubleshooting process.

Changes in Security Configuration

Host-based IDS also have the capability to recognize when a change has been made to the system security configuration. This feature can assist in identifying the precursor to a planned attack. A proper security configuration is critical to stopping both insiders and outsiders from misusing your computer systems. Securing a system normally happens at one of two times, when the system is built, or when the system is deployed onto the network. The wise thing to do as a network security administrator would be to accomplish this when the system is built so that there is no window of opportunity to exploit the system. Such as if the system is deployed on the network, and before being secured is inadvertently stepped away from, and during that interim, the system is compromised.

Availability

Another beneficial feature of the host based ID is that it possesses the capability to work even if encryption is being used. This is accomplished due to the fact the software resides on the "receiving" system, and the packet is decrypted upon arrival at the system. This allows a host-based system to check the packet for any possible attack signatures, where as a network-based ID system may not be able to read the packet due to its encrypted contents.

In addition to these features, a host-based ID system can possibly identify errors in configuration that may have been missed when the system was being setup or deployed. This offers an additional layer of security so that the administrator can be notified and resolve the potential problem before it becomes one.

Challenges of Host-Based ID Systems

All systems have challenges, where they do not excel, or cannot perform under certain conditions. For these reasons implementing both network-based and host-based IDS is your best method for intrusion detection. Both types of systems complement each other. Both systems have downfalls and in combination with the many different security

technologies, we can secure our networks, assets, data, information, or resources under our care.

Earlier, it was mentioned that every IDS has some drawbacks or limitations that come along with them. This is no different with the host-based IDS.

One of the limitations of host-based IDS is that is it usually confined to the individual computer on which the software has been loaded. Host-based ID systems are reliant on the host system for them to perform up to par. This means the performance of the host system will directly reflect on the quality of the data it gathers or attempts to gather as the case may be. The host-based IDS software is reliant on the host computers memory, CPU, network adapter and other factors that contribute to the overall performance of the system. As stated in Chapter 4 of the Practical Intrusion Detection Handbook "A Windows NT workstation will generate about 1Mb of data per day in logs, a Windows NT Server about 10Mb, a Unix workstation about 8Mb, and a Unix Server about 20 Mb". If we combine this into a standard network of say 50 hosts mixed between Unix and Windows and 10 servers mixed, you will quickly notice how the traffic adds up to almost 400 Mb of data per DAY!" this quickly degrades the performance of most systems.

Another issue that can arise with host-based ID systems is the deployment method. This presents a problem since each target system must have the agent installed on it. Initial deployment and maintenance require either an automatic deployment method and update plan in place or having someone manually run around and install the application, which is not a completely feasible method. If one of these is not accomplished the host-based agent signatures can become quickly outdated.

Since we have already identified that host-based agents are on the target system that they monitor. This presents the situation where if an unauthorized user gains access to the system they can possibly disable the agent. This would render the IDS system useless and blind to what is going on that system. This activity should raise at least some suspicions if you see a system or multiple systems suddenly go offline. This is like someone suspicious covering your eyes and ears and asking what they are doing? Would you not be curious as to what they are up too?

Spoofing presents another issue that can be difficult to handle for the host-based IDS system. Spoofing a host-based system can be accomplished by inserting false records into the audit stream that indicate false or non-existent activity. A good protective measure to defend against this would be to have a reliable, trustworthy, protected audit source.

A limitation with some of the tools located on the host system such as with netstat, lsof, top, as well as some others, is that the administrator must be watching for the tools to be effective, as well as to be classified as an IDS system.

Network-Based Intrusion Detection Systems

Network-based intrusion detection systems are used in the analysis and detection of malicious packets. This differs from the host-based since they do not examine log files and do not concern themselves with the "individual" system; rather they look at the traffic on the network destined for the system(s). The method used to see those packets is called "packet capturing" or "sniffing" the packets, in some cases this can be derived from the output of network devices such as routers or switches.

Network IDS systems are normally closer to the originating source of the packet than the destination. This allows for a different standpoint on access attempts, Denial of Service (DoS) attempts, and maliciously crafted packets, especially those originating from external to your network. In essence, the network based id system has the better perspective on packets coming in from the outside of your network than does the host-based system which will protect systems and offers the better perspective of internal points of interest.

Benefits of Network-Based Intrusion Detection

There are many benefits to the network based intrusion detection systems. While they do not protect the individual host system, they are capable of covering a large area and monitor network traffic arriving into and departing from the monitored network. While working hand in hand with host-based systems, the network-based and host-based systems provide the capability to recognize and prevent, identify, and possibly resolve most computer misuse. In addition, network based intrusion systems benefits include external threat deterrence, identification, and automated response capabilities.

External Threat Deterrence

Hackers know that the presence of an intrusion detection system makes it more likely they will be identified or caught. This makes it very risky for hackers since they could be caught in the act attacking the system or network. If this occurs, they can possibly face criminal or civil punishment.

A beneficial way to enhance the deterrent value is to respond to an attack with a follow up email or phone call, or even reporting the case to authorities. Spoofing of the source address is unfortunately very common and may decrease the deterrence value.

Detection

Network IDS systems have proven themselves to be valuable in identifying threats to your network and to assist in troubleshooting. By troubleshooting, what is meant is that a network id system can assist in a statistical analysis of traffic, by identifying what is normal and what is outside the norm for that network. When traffic arrives outside the norm, it is easily identified in the statistical analysis of the traffic. In addition, the signatures are able to determine from the traffic that they check against whether the traffic is appropriate or not.

Automated Response/Notification

Network-based ID systems can be setup and configured for automated response or notification when there is an alert or traffic of concern. Some examples would be to have an email sent to the administrator of a system to let them know that their system is being targeted by suspicious traffic. This would put the system administrator in a higher state of alert for that system while a response is being implemented to stop the traffic, stop the attack, or monitor for subsequent activity and follow-up.

Some **Auto-response** options would be:

Automatically place ACL's: Some Network IDS systems can automatically place rules into the firewall or router to block offending traffic.
Defensive Attack/Attack Back: The IDS could respond with a attack of its own back to the offending host in hopes you take them offline before they take you offline. This scenario should be VERY closely scrutinized and probably should not be used due to legal reasons.
Break the Sequence/Reset: The network-based IDS can break the TCP Sequence by spoofing a packet to both systems sending a RST causing the connection to be reset and hence broken. This can stop attacks in mid-attack.

Some **notification** options available are:
Pager: Have Alert sent to pager so you can be notified no matter of your location.
SNMP Traps: Usually this option is used to notify Security Operations Center or Network Operations Center of incident.
Console: Places the alerts on the console of the IDS system. (Requires someone watching the system to determine attacks.
Audible: Causes a sound file to be played when an attack occurs. – (Requires someone to listen for attacks)
E-Mail: Sends an email to distribution list or specific individual when an alert occurs.

With network-based IDS, in many cases there is the high probability that a host-based IDS may not even detect, or even may be impossible for, the host-based IDS to detect the incoming network traffic attack. If a network-based ID system is not in place, the attack may go unidentified (not meaning unaffected). Perhaps once the person has gathered the "recon/intel" information from the system and gathered access to the system we would know, but this is a little too late in terms of security. A network-based ID places a special notification and possibly a protective barrier in place. It allows us to identify threats preemptively and identify areas where an attack may be targeted.

In addition to this a network based ID system allows us to cover areas in network security such as Confidentiality, Integrity, and Availability.

Confidentiality

Confidentiality of information should still be possible even over the network. The information should be kept confidential via access measures such as login authentication.

Unauthorized Access

Unauthorized access over the network should not be possible. An access mechanism should verify accessibility prior to granting access to systems over the network. Due in part to the security vulnerabilities located in some software tools and applications used to share that information there are exploits that can allow unauthorized access to the network. Take for example tftp or wu-ftpd there are many exploits out there that can take advantage of security holes in these and other programs.

Staging Grounds/Jump-Off Point

Under many circumstances, a system is compromised from another system that has already been compromised. This can by identified as a staging ground attack. The system is very rarely targeted from the perpetrator's home system. Usually they use a maze of systems to throw off the path from which they arrived. By targeting and receiving access to the single system they can than use that as their "staging ground" for further attacks outside the organization or additional systems within the organization, further throwing off the trail to the origin of the attack. For example, why would a "PBX system" have "anonymous ftp traffic"? This activity is suspicious and should be investigated or closely monitored.

Downloads of Unauthorized/Security Files

With the network-based IDS in place, you can identify traffic that has potentially harmful strings in the packet payload. Such as inside of FTP traffic you see /etc/passwd or /bin/sh (or one of the other shells). You know that there is something amiss. This was and still is a very common vulnerability. The traffic that matches this pattern should be noticed. This activity can indicate very critical information is being "gathered" or at least attempted to being gathered.

Integrity

Integrity of systems is critical to their "survival". With network-based IDS in place, the capability to identify malicious or harmful traffic before it arrives at the host is possible. The integrity of your network traffic can be "sanitized" so to speak before arriving at the host. This ensures or hopes to ensure the data is clean from potentially harmful data.

Suspicious Traffic To/From Ports

With the Network-Based IDS, it is possible to identify traffic directed into your network and destined towards ports on the system, such as with the latest SNMP exploits, or with the Code-Red Virus. For example, in the Code-Red Virus attack, while a host-based IDS would be able to detect it on the individual system with the network-based IDS you can see which systems are broadcasting the traffic halting the source hopefully and identify on a "higher level" overview which systems could potentially have been affected. Alternatively, in the case of Back Orifice a network-based IDS can detect the traffic and if setup can halt or drop the attack, so it does not affect the end system.

Availability

Availability of systems is critical to a majority of organizations success. In some cases when a system is offline, the organization is losing money. Take for example, the case in which E-Bay was taken offline due to a Denial of Service, attack. During the outage, they were not conducting business, which in turn affects the bottom line of their main business. Damages in the past have cost in the range from hundreds to hundreds of thousands of dollars in lost business, and productivity. Unfortunately, the real loss is very difficult to quantify and often goes unreported.

Denial of Service (DoS/DDoS)

Denial of service attacks sole purpose is to remove a service, computer, or other resource offline so that it cannot be accessed for normal usage, or possibly as a distraction method. Denial of Service attacks come in a variety of levels varying in severity and intensity. These attacks can be directed and slow down the targeted network or the network from which they are directed to a crawl. This causes their own set of problems for the business units that are affected by the DoS attack. DoS attacks can originate from inside your network or outside your network. The packets that make up these attacks have certain characteristics about them that make them easy to identify for Network-Based IDS systems. There are three main forms of Denial of Service attacks, Malformed Packets, Packet Flooding, and Distributed Denial of Service.

Malformed Packets

Malformed packets are packets that have been modified to deviate from the standard packet protocol. They attempt to evoke a response from the receiving system that causes the system to go offline or the protocol stack to crash. This activity can cause a variety of responses up to and including systems to crash completely. Such as the case with a Microsoft 9X problem occurred when you connect to port 139 and the urgent packet was set and data directed at the system. This caused the system to lock up and/or blue screen, forcing the user to hard boot the system. This could cause unexpected data loss and other issues that affected the availability of the affected systems.

Packet Flooding

This type of Denial of Service is relatively simple and can be extremely effective. This involves sending as many packets as you can at a network device until it crashes or becomes unreachable. This type of attack send so much data the device slows to a halt or becomes so slow that valid users and usage cannot traverse the network through this device. If the user is valid, tracking them down can be very easy though if the packets have originated from a spoofed location tracking to the source can be very difficult if not impossible. For example: A spoofed SYN Flood that is targeted at a network resource can cause the resource to become unavailable for normal usage. This is accomplished due to the way SYN packets are handled the system must respond back with a SYN/ACK as the second step in the three-way handshake. The SYN/ACK will be sent and wait for a response from the "initiating" system. As it waits for the response from the "initiating" host, it will not receive one since the "spoofed" host never initiated the communication, or it doesn't exist.

Distributed Denial of Service (DDOS)

This activity involves a similar tactic as the packet flooding except on a much larger scale and can be very difficult to halt. By a larger scale, what is meant is that many computers are used to target a single device to halt its operation. If the IP is spoofed, it is very difficult to block or ignore the offending traffic. In many cases of DDOS the attacking computers are unaware zombies. Normally these systems were "recruited" by virus programs or Trojans from the internet. These zombie systems lie in wait for a

command from a console to attack a certain target. Network IDS systems cannot protect against these types of attacks but can assist in detection and response.

Traffic Analysis Signatures

Traffic Analysis Signatures are sometimes also called Network Signatures. ID systems use these to verify packets based upon content or flags. These signatures can be made up or any combination of patterns in the data. Can you imagine if you had to analyze this traffic manually? If it was on a 100Mbps link the time to analyze and identify threats would not be responsive enough to identify the problem and respond to it.

Additional Uses

Surveillance

Another of the valuable uses of network-based IDS would be for investigation and surveillance use. With the ability to filter and monitor traffic, you can closely monitor a specific traffic or pattern that is known. Normally this information is gathered from the traffic patterns or another forewarning. In addition, this information can be used for forensic analysis.

Forensic Usage

The Network IDS system can be used to gather and analyze network traffic. Essentially the same tools that were used during the surveillance and investigation phase are available to be used for forensics analysis. This allows some additional uses such as:

- Monitoring online transactions
- Track network growth
- Generate details of how your network services are being used
- Identify unexpected changes in network or its behavior

Challenges for Network-Based IDS

As with the host-based systems, there are challenges for network-based IDS systems. Some of these challenges can be eliminated with the use of host-based and network-based together as a team. The implementation of both systems is highly recommended so that the benefits, protection, and coverage that are received are more comprehensive.

One of the areas that network-based IDS systems have a problem is with encrypted traffic. The reason for this is that the network-ids can identify it is encrypted but it is unable to read the packet data or the underlying information within the packet. Encryption is becoming increasingly used throughout multiple layers of the OSI model.

Another of the challenges for network-based ID systems is the use of high-speed connections. The reason is that the information being parsed by the IDS just doesn't have the capability to keep up with that traffic level. This causes packet-loss and this is an issue for some network-IDS systems. It can cause packets to not be seen and so the full signature of an attack may not be seen. This would cause no reporting of a valid or possible attack.

Similarities between Network/Host based IDS

One of the similarities of both network and host based ID systems is that they look for "Attack Signatures". Attack Signatures are a specific set of patterns that can identify unscrupulous or malicious intent on the part of the attacking person/system. Host-based IDS tend to look for attack signatures in logs or event files. A network-based ID tends to look at network traffic and compare it to a signature database/rule set.

Both types of systems can be used to cover their respective areas involving confidentiality, integrity, and availability of systems. While they each cover their individual areas such as network-based IDS will focus on network packets and traffic, and the host-based system will focus on the log files or traffic directed at that host, they work together to identify potential threats to network resources.

In addition, both systems provide an added layer of security to the network they are protecting. This is accomplished since it gives the administrator "eyes in the field". What this means is that the administrator does not have to manually check each system; both technologies have the capability to alert the administrator of potential problems via notification methods, if an alert is generated or problem identified.

Finally, when both systems are working together you have the best chances of identifying potential problems, since you will have a network layer of defense as well as a host-based defense mechanism in place. In the situation, you only have one layer of defense you may miss the attack that is directed at your system. The added benefit of host and network based on the IDS method you are able to identify both types of alerts directed at your host as well as potentially troublesome network traffic.

Sources

1. Sans Institute Resources
   Intrusion Detection FAQ- What is host-based Intrusion Detection
   by Laurie Zirkle
   http://www.sans.org/newlook/resources/IDFAQ/host_based.htm
2. Intrusion Detection
   by Andrew Conry-Murray, Network Magazine-Dec 5, 2000
   http://www.networkmagazine.com/article/printableArticle?doc_id=NMG20001130S0007
3. The Evolution of Intrusion Detection Systems
   by Paul Innella, Tetrad Digital Integrity, LLC November 16, 2001
   http://online.securityfocus.com/infocus/1514
4. The Practical Intrusion Detection Handbook,
   by Paul Proctor, Prentice Hall, 2001
5. An Introduction to Intrusion Detection & Assessment for System and Network Security Management
   by Rebecca Bace from Infidel, Inc. for ICSA Inc.
6. Intrusion Detection: Reducing Network Security Risk
   by Recourse Technologies
7. Sans Institute Resources
   Intrusion Detection FAQ –What is network-based Intrusion Detection
   Stephen Northcutt, SANS Institute
   http://www.sans.org/newlook/resources/IDFAQ/network_based.htm
8. Network- vs. Host-based Intrusion Detection
   A Guide to Intrusion Detection Technology
   Internet Security Systems, Oct. 1998

Assignment 2 - Network Detects

Method for Gathering Data

All data gathered in this assignment was gathered from one of the following network connections and configurations. In the initial configuration, this is labeled, as "home-network". The systems are configured as in Fig 1.1

Snort is configured on both the Home Network and the Work Network. The "home-network" is using Snort v1.8.3 for Windows using Politecnico di Torino Packet Capture library. Snort has a few rules that I have written to track all traffic directed at certain ports on the network and specific systems.

Snort Command Lines

Windows: (System from Fig 1-1):
C:\Progra~1\IDScenter\Snort.exe -c C:\Progra~1\IDScenter\snort.conf –l D:\snortlog -A full -h 198.xxx.xxx.192/27 -a -b -C -d -e -X -U –y

FreeBSD/Unix: (System from Fig 1-2):
snort -c /usr/local/etc/snort.conf -l /var/log/snort -A full -h 216.xxx.xxx.128/26 -a -b -C -d -e -X -U -y –D

Snort	–l <directory>	- lists the Snort Directory where the logs are placed
	-A Full	- lists the Alert mode Snort in which snort is placed. For this practical, Full Alert mode was used to ensure the comprehensiveness and gathering the header of the packets.
	-h <network>	- sets the "home_net" or home network – this must be in 198.XXX.XXX.0/24 notation
	-a	- displays ARP packets when packets are decoded
	-b	- sets the mode for logging packets in binary mode – this mode is faster and more efficient for logging than ascii, since no conversion is necessary to convert from the binary to ascii format
	-C	- Prints the character data from packet payload
	-d	- dumps the application layer data when displaying in verbose or packet logging mode
	-D	- Tells Snort to run as a Daemon
	-e	- display the link layer packet headers to allow for easier troubleshooting and analysis
	-X	- Dumps the RAW packet data starting at the link layer
	-U	- Changes the timestamp in all logs to be in UTC
	-y	- Include the year in alert and log files

Fig. 1

Home Network

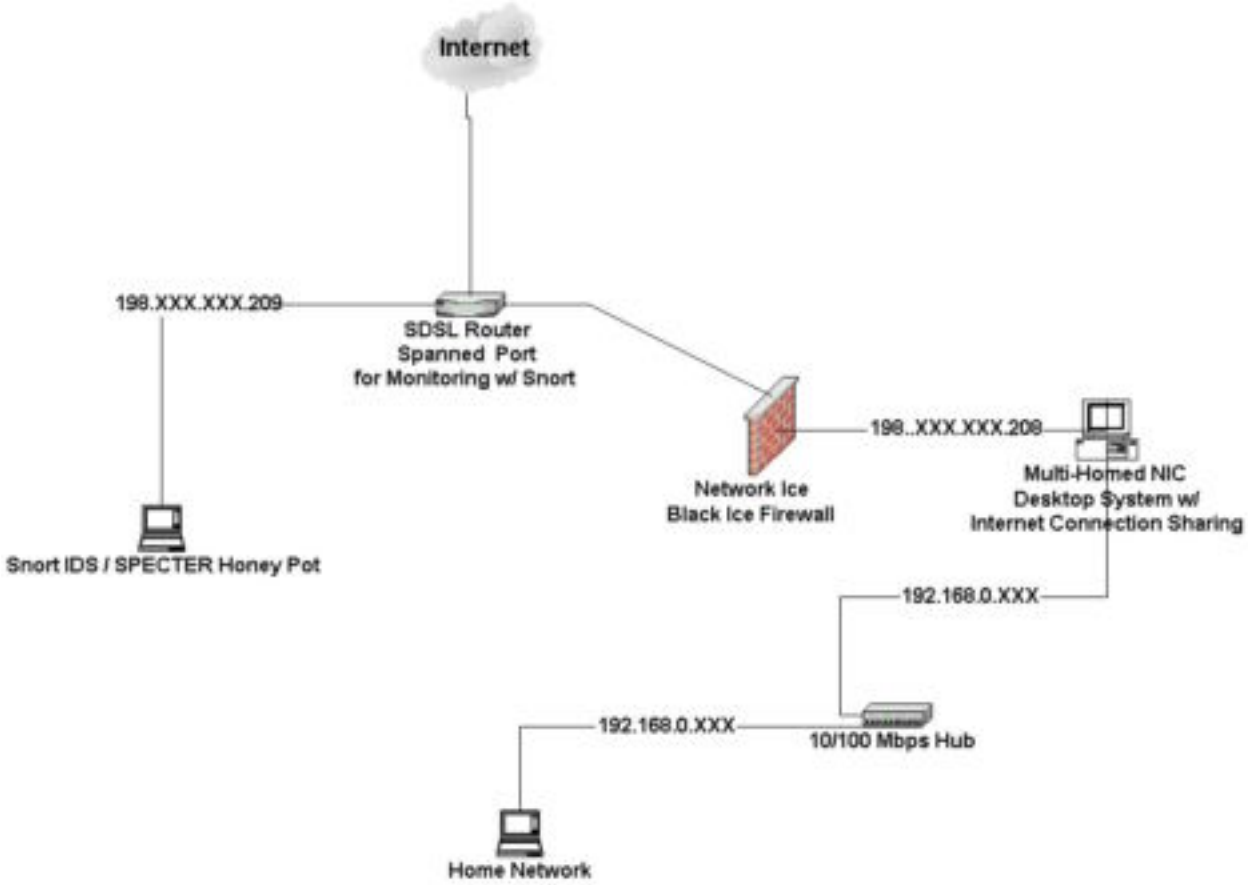

Fig 1.1

Work Network

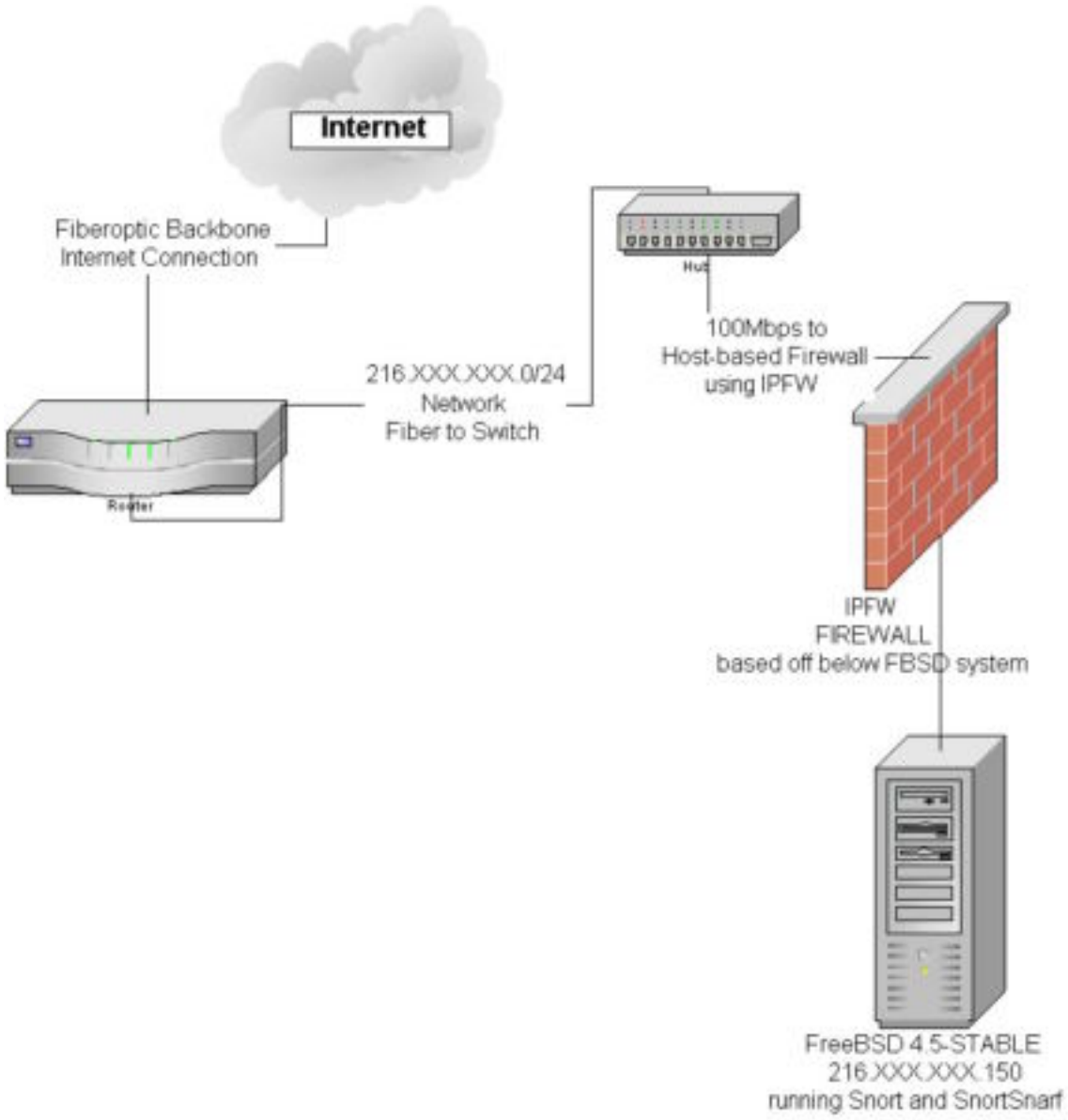

Fig 1.2

Alert 1

SNMP UDP/1993 Scan

Mar 18 08:28:15 142.165.148.253:39812 -> 216.XXX.XXX.150:1993 UDP
Mar 18 08:28:15 142.165.148.253:39813 -> 216.XXX.XXX.151:1993 UDP
Mar 18 08:28:16 142.165.148.253:39822 -> 216.XXX.XXX.160:1993 UDP
Mar 18 08:28:16 142.165.148.253:39823 -> 216.XXX.XXX.161:1993 UDP
Mar 18 08:28:16 142.165.148.253:39824 -> 216.XXX.XXX.162:1993 UDP
Mar 18 08:28:16 142.165.148.253:39825 -> 216.XXX.XXX.163:1993 UDP
Mar 18 08:28:16 142.165.148.253:39826 -> 216.XXX.XXX.164:1993 UDP
Mar 18 08:28:16 142.165.148.253:39827 -> 216.XXX.XXX.165:1993 UDP

Source of Trace

The source of the above network trace was derived from the FreeBSD 4.5-STABLE server on the "work-network". Reference Fig 1-2

Detect was Generated by

This detect was generated using Snort v1.8.3 for FreeBSD. The results for this were parsed by SnortSnarf on the same FreeBSD system. Snort identified this merely as a UDP port scan though upon closer examination it shows it was a possible probe seeking Cisco Routers with SNMP running.
The rule that identified this probe was:
preprocessor portscan: $HOME_NET 4 3 portscan.log

Probability the Source Address was spoofed

Spoofing is highly possible since this is a UDP scan though it is unlikely. The reasoning behind this is that the attack is very recent and it scans multiple hosts on the network seeking a reply. An attacker would want to receive this information back as recon information for possible future assault.
Also take notice of the closely incrementing port numbers from the "probing" system. They are all sequenced which further indicates this is a legitimate probe.

With this evidence presented, the conclusion drawn is that this is a probe from the attackers system or another compromised system.

Description of Attack

This attack has the high probability that it is searching for possible vulnerable Cisco Routers using SNMP. This is determined by the port they are searching and its associated service that it provides. Also taken into account is the CERT publication that was recently published.
http://www.cert.org/advisories/CA-2002-03.html
http://www.sans.org/alerts/SNMP.php

Both of these URL's list advisories in regards to a SNMP attack, and 1993/UDP

Attack Mechanism

Was this attack a Stimulus or Response?
This scan of UDP port 1993 was initiated by an outsider and was the initial stimuli that started the attack. Due in part to the fact that this is a UDP packet there is no 3 way handshake that will assist in determining whether it was stimulus or response it must be shown through another method.

Questions asked to determine Stimulus or Response

Are there any services that communicate on this port, which would have drawn this type of traffic?

No, this is shown with the command "sockstat –l | grep 1993" on the host system which lists the listening ports that match 1993 on the local system. No open ports were located to match the criteria.

What Service is being targeted?

The service that is being targeted was SNMP for Cisco.

Does the service have known vulnerabilities?

Yes, the SNMP for Cisco could be affected by this exploit. However, this system would not have been affected by this exploit. This FreeBSD system would not have had the specified port listening.

Is this benign, an exploit, denial of service, or reconnaissance?

This attack would classify as an exploit/recon attack. It appears the attacker attempted to target the entire network perhaps seeking to exploit routers or other vulnerable systems to this SNMP exploit.

The attack mechanism used could be a variety of tools up to and including a new scanning tool released by Foundstone. This tool is normally used to identify areas in your own network where their maybe problems but of course there are those out there who will use it for malicious intent.
The tools listed below are able to duplicate the traffic that was generated.
http://www.foundstone.com/knowledge/infoterms.html?filename=snscan.zip
or
http://www.sans.org/snmp/tool.php

Correlation

This vulnerability was identified by has been identified by Oulu University Secure Programming Group. CERT brought the vulnerability to the attention of the information security community. In addition to notifying the community, they are working with vendors to have patches released.

http://www.cert.org/advisories/CA-2002-03.html

Evidence of Active Targeting

Look at the below trace, from **Fig. 6-1,** there is significant evidence to show that the attacker was targeting this network.
All the source ports from the attackers system are in a sequential order. The IP addresses that were not monitored, or did not respond, correspond with the sequencing of the attackers ports, to indicate a SYN type scan.

Example:
Attacker port 39812 correlates to IP address 216.XXX.XXX.150
Attacker port 39813 correlates to IP address 216.XXX.XXX.151
SKIP for NON-RESPONDING OR NON MONITORED HOSTS
Attacker port 39822 correlates to IP address 216.XXX.XXX.160

Mar 18 08:28:15 142.165.148.253:39812 -> 216.XXX.XXX.150:1993 UDP
Mar 18 08:28:15 142.165.148.253:39813 -> 216.XXX.XXX.151:1993 UDP
Mar 18 08:28:16 142.165.148.253:39822 -> 216.XXX.XXX.160:1993 UDP

```
Mar 18 08:28:16 142.165.148.253:39823 -> 216.XXX.XXX.161:1993 UDP
```

Fig. 6-1

This shows the attacker was scanning the entire subnet or at least this closely related range of IP addresses in this class. This provides almost conclusive evidence of active targeting of the systems on this network.

Severity of Attack

(Criticality + lethality) – (system + network countermeasures) = severity

Criticality of System = 5
The active targeting of routers can affect routing and lead to Denial of Service of multiple systems.
Lethality = 5
Due to the wide spread use of SNMP this should be rated a 5

System = 5
This attack on this system will not work due to the fact this is a FBSD system and the Cisco Routers normally use this port not host systems.

Network Countermeasures = 5
This system has IPFW, which is blocking SNMP from outside hosts, as well as the current updates for SNMP are in place. This attack would not likely succeed against this Host. In addition, this host has nothing communicating on that port. With this in mind Network countermeasures should be set to 5.

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

(5 + 5) – (5 + 5) = 0

This system rates as a 0 severity.

Protection Measures

Some measures that should be implemented to protect the additional systems on your network would be to; Block SNMP at the router from outside sources, Specify ACL lists on the routers to protect your systems and take appropriate precautions listed from the vendor of your system. In addition, ensure that all patches are in place for the systems.

Test Question – Alert #1

Which of the following choices is **not** able to be accomplished by a host-based IDS system?

A. Detect Malicious Traffic Directed at the closest router
B. Protect files on the system protected by the host-based IDS
C. Protect host from unauthorized access
D. Offer protection from attacks directed at the host

Answer: A
A host-based ID system cannot protect a system other than the system on which it resides. Example: A Windows NT system is protected by BlackICE Defender. It cannot provide protection for a Windows XP system on the same network.

Alert 2

SMTP Connection/Scan Attempt

Mar 21 00:04:15 218.24.129.19:1694 -> 216.xxx.xxx.150:25 SYN ******S*
Mar 21 00:04:15 218.24.129.19:1696 -> 216.xxx.xxx.151:25 SYN ******S*
Mar 21 00:04:15 218.24.129.19:1698 -> 216.xxx.xxx.160:25 SYN ******S*
Mar 21 00:04:15 218.24.129.19:1700 -> 216.xxx.xxx.161:25 SYN ******S*
Mar 21 00:04:15 218.24.129.19:1703 -> 216.xxx.xxx.162:25 SYN ******S*
Mar 21 00:04:15 218.24.129.19:1707 -> 216.xxx.xxx.163:25 SYN ******S*
Mar 21 00:04:15 218.24.129.19:1710 -> 216.xxx.xxx.164:25 SYN ******S*
Mar 21 00:04:16 218.24.129.19:1714 -> 216.xxx.xxx.165:25 SYN ******S*

Fig 8.1

Source of Trace

In figure 8.1 there is a scan attempt of port 25 (SMTP). This scan was taken from the FreeBSD 4.5-STABLE system located in Fig. 1-2.

How was the Detect Generated?

This detect was gathered using Snort 1.8.3 and parsed with SnortSnarf. Out of the normal traffic that arrives at the SMTP port, this was identified as a simple portscan by Snort due to the time sequence in between the connection.
The rule within Snort 1.8.3 that identified the probe was:
preprocessor portscan: $HOME_NET 4 3 portscan.log

Probability the Source Address was spoofed

While there is a slight chance this was spoofed, it is highly doubtful. With this attack, there doesn't appear to be a full connection accomplished by the attacking host, but this has the earmarking of a SYN probe for perhaps an open relay, misconfigured mail server, or for recon purposes to determine OS, mail server version, and other possible critical information that could be used in a attack at a later date. In addition, SMTP is a valid port on this system, and many times is misconfigured and left as a open relay, it appears there is a high chance the attacker would probably want to return and verify the findings of this recon attack. (Which did occur shortly after the attack and scanned than for a Proxy)

Description of Attack

This attack that was captured was very possibly accomplishing a recon for misconfigured mail systems to use as a Open Relay for SPAM (not the food but unsolicited email), or as a attempt to recon the mail server and see what mail service it is running. Many mail systems run as a privileged user and if buffer overflowed can lead to those permissions. In addition, many mail systems announce the type and version of mail system when you connect to that port. This information provides valuable reconnaissance information to an attacker. Consider Fig. 9.1 (section 2) from the information gathered you can tell that it is a Windows System (Windows NT, 2K,XP). You can tell the version of the mail server running, as well as with certain command such as EXPN and VRFY they will give responses for any valid accounts on the system. This poses critical security risks for the system if the OS is already known.

Fig. 9.1

This is a properly configured mail server response:
root [~]: telnet myhost.net 25
Trying 216.xxx.xxx.162...
Connected to myhost.net.
Escape character is '^]'.
220 union.myhost.net ESMTP ; Wed, 20 Mar 2002 18:11:16 -0800 (PST)

This is misconfigured mail server response:
root [~]: telnet smtp.myisp.com 25
Trying 4.abc.def.ghi...
Connected to smtp.myisp.com.

Escape character is '^]'.
220 exchange.myisp.com ESMTP Server (Microsoft Exchange Internet Mail Service 5.5.2653.13) ready

Attack Mechanism

Was this attack a Stimulus or Response?
The scan swept across multiple hosts looking for a specific port. This connection attempt to port 25 of the system would be considered stimulus. This is due in part to the initial SYN connection. Also normally for the SMTP service this is where incoming mail normally arrives and where users communicate with the service to tell it what to do. For example in **Fig. 9.2** demonstrates a normal communication session with a SMTP server.

Fig 9.2

tcpdump -s 1514 'dst port 25' > SMTP.traffic.txt
21:43:10.979137 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: S 661955649:661955649(0) win 16384 <mss 1460,nop,nop,sackOK> (DF)
21:43:11.129444 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: . ack 1345033469 win 17520 (DF)
21:43:11.454210 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: P 0:18(18) ack 70 win 17451 (DF)
21:43:11.605741 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: P 18:53(35) ack 136 win 17385 (DF)
21:43:11.781546 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: P 53:88(35) ack 183 win 17338 (DF)
21:43:11.943205 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: P 88:94(6) ack 235 win 17286 (DF)
21:43:12.269462 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: . ack 258 win 17263 (DF)
21:43:12.368112 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: P 94:100(6) ack 258 win 17263 (DF)
21:43:12.519359 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: P 100:135(35) ack 281 win 17240 (DF)
21:43:12.693649 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: P 135:170(35) ack 328 win 17193 (DF)
21:43:12.851412 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: P 170:176(6) ack 380 win 17141 (DF)
21:43:13.012067 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: P 176:845(669) ack 430 win 17091 (DF)
21:43:13.256559 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: P 845:850(5) ack 430 win 17091 (DF)
21:43:13.494129 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: P 850:856(6) ack 484 win 17037 (DF)
21:43:13.495471 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: F 856:856(0) ack 484 win 17037 (DF)
21:43:13.644116 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: R 661956507:661956507(0) win 0 (DF)
21:43:13.645492 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: R 661956506:661956506(0) win 0
21:43:13.647124 198.xxx.xxx.208.4637 > union.myworkhost.net.smtp: R 661956507:661956507(0) win 0

Questions asked to determine Stimulus or Response

Are there any services that communicate on this port, which would have drawn this type of traffic?
The answer to this is yes, the targeted system does have a SMTP server located on it. Since the answer was yes, could this traffic be legitimate traffic? The answer can be yes or no. Since the question above can be answered with a yes or no response, further investigation is necessary to determine the intent or nature of the traffic.

What service is being targeted?
SMTP, or Simple Mail Transfer Protocol, this service provides mail transfer and operates on port 25

Does the service have known vulnerabilities?
While there are vulnerabilities, it depends on the operating system running the SMTP service. So the answer can be Yes and No and depends on the configuration by the Administrator. There are vulnerabilities in many of the releases out there of SMTP. The possible vulnerability in this case, can mean use for unauthorized mail sending, sending of viruses, and/or other malicious mail services such as mail bombing or critical issues that can arise from mail servers run amuck.

Are there attack tools present for this service?
Yes, there are attack tools present across the internet to search for open mail relay systems. These open mail relays are frequently used for SPAM and for sending mail from alternate hosts.

With the answers to these questions, it appears the traffic is a Stimuli and not a response.

Correlation

Recently, there has been similar traffic being scanned on various networks including the "Home-Network" which resides on a completely different subnet/network block and completely different upstream providers. In addition, Mail abuse is appearing to be on the rise.

In addition, reference the following log files from the honeypot system. Located in **Fig. 11.1 and 11.2**

Fig. 11.1

```
************************************************************
SMTP connection
Host  : 205.251.246.113  (wiley-1-428186.roadrunner.nf.net)
Domain :
return address  :
target address :
Time : Mon Mar 18 13:29:02 2002
Log :
Client connecting: 205.251.246.113
--->220 system.honeypot.net WindowsNT SMTP server 3.1.7 at Mon Mar 18 13:29:03 2002
Connection timed out
Closing connection with 205.251.246.113
```

Fig 11.2

```
************************************************************
SMTP connection
Host  : 63.144.237.193
Domain :
return address  :
target address :
Time : Sun Mar 17 07:32:20 2002
Log :
Client connecting: 63.144.237.193
--->220 system.britsys.net Microsoft ESMTP MAIL Service, Version: 6.0.2600.1 ready at Sun
Mar 17 07:32:25 2002
Connection timed out
Closing connection with 63.144.237.193
```

Evidence of Active Targeting

In order to determine if Active Targeting is taking place we must ask at least these three questions.

Are they targeting a specific host?
NO
In the evidence provided, in **Fig 8.1** it appears the attacker is scanning the entire network seeking for SMTP mail systems. This is proven by the fact that they have scanned more than one IP address and the sequencing tends to stay in line with that network.

Is this a general scan of entire network?
YES
According to **Fig. 8.1,** there is evidence to support that the attacker was scanning the entire network and not just a single host or random hosts. This is shown by the time in between the scan activity and the corresponding hosts as well as the source ports from the attacker system. They are close together to indicate that the attacker was scanning each block at a time.

Is this a probable "wrong number"?
NO
With the evidence provided this eliminates the option of a wrong number as SMTP actions do not scan networks looking for additional SMTP mail servers send mail from. With the programs that use SMTP normally, the settings are specified by the operator of that system. The SMTP server is normally specified by your upstream provider or other source.

The conclusion drawn in the above alert is that this was not a case of Active Targeting of the system but a network scan of the entire network.

Severity of Attack

(Criticality + lethality) – (system + network countermeasures) = severity

Criticality of System = 5
The system being attacked is used for mail delivery as well as additional web services. If this system were to be compromised, it would represent a significant problem with customer communication. This makes this system critical if it was compromised by an attacker.

Lethality = 2
Currently there are few if any exploits available for this version of SMTP running on this system. (Sendmail) There are some exploits available for older versions as well as there, being possible unreported exploits out in the wild. In addition since this is appears to be a probe it is very unlikely to be lethal to the system at this point. These factors would have the rating of the Lethality at two.

System Countermeasures = 5
This system has Open Mail Relay turned off. In addition, there is a file, which allows only specified hosts the ability to relay from this system. In addition, this system has been tested by third Party software to determine that the SMTP has been secured properly. With these factors in place, System Countermeasures are at four.

Network Countermeasures = 1
This system has IPFW installed but we are unable to block port 25 due to the fact all mail systems communicate on this port, and inbound mail would be hindered if this port was blocked or filtered. With this in mind Network, countermeasures should be set to one.

With this in mind

(Criticality + lethality) – (system + network countermeasures) = severity
$$(5 + 2) – (5 + 1) = 1$$
With the formula, it shows that this factor is one. While this shows some concern for the activity we should check to ensure that the system is secured from unauthorized use.

Defensive Measures

In order to better protect your mail server there are a few steps that should be accomplished. The server should be configured to ensure that the system couldn't be used as an Open Mail relay. There are websites such as the MAPS (Mail Abuse Prevention Service) website that offer suggestions on how to protect each individual mail system. In addition, ensure all patches are up to date and installed. In addition, in order to remove the reconnaissance portion, the administrator should alter the headers of the mail system. For instance, the examples listed in **Fig 11.1 and Fig 11.2.** This action can force the attacker to make a more concerted effort of bypass your system altogether, since they would be forced to work to get Operating System, Mail Server version, and possibly other critical information.

Test Question Alert #2

Which of the following actions can a administrator accomplish to prevent reconnaissance data from being passed via a SMTP server?

A. Modify the configuration files so that the EXPN and VRFY commands do not work
B. Add all hosts to the relay allow files access file
C. Add ONLY known hosts to the relay allow access file
D. Remove Operating System and version information from the SMTP banner

Answer:

A, C, D

In order to properly secure mail servers the administrator must restrict access to the mail service so it does not allow everyone to send email from the service. This right must be restricted to "trusted" users. The EXPN and VRFY commands allow

Answer (cont.)

attackers to gather potential user data from the server, these should be disabled by the Administrator to further protect the system. Removing the non-essential banner information makes it harder for the attacker to gather information from the SMTP server. The harder they have to work the less chance they will target your system.

Alert 3

Two Different Proxy Scans from the same Source IP

Mon.	Day	Time	Source	Src Port	Destination IP	Dest Port		PCKT FLAGS
Mar	22	7:07:01	65.16.184.131	2006	-> 216.XXX.XXX.150	21	SYN	******S*
Mar	22	7:07:01	65.16.184.131	2008	-> 216.XXX.XXX.150	25	SYN	******S*
Mar	22	7:07:01	65.16.184.131	2009	-> 216.XXX.XXX.150	80	SYN	******S*
Mar	22	7:07:01	65.16.184.131	2010	-> 216.XXX.XXX.150	110	SYN	******S*
Mar	22	7:07:01	65.16.184.131	2011	-> 216.XXX.XXX.150	119	SYN	******S*

Mar	22	7:07:01 65.16.184.131	2012	-> 216.XXX.XXX.150	1080	SYN	******S*
Mar	22	7:07:02 65.16.184.131	2013	-> 216.XXX.XXX.150	6588	SYN	******S*
Mar	22	7:07:02 65.16.184.131	2014	-> 216.XXX.XXX.151	21	SYN	******S*
Mar	22	7:07:02 65.16.184.131	2016	-> 216.XXX.XXX.151	25	SYN	******S*
Mar	22	7:07:01 65.16.184.131	2017	-> 216.XXX.XXX.151	80	SYN	******S*
Mar	22	7:07:01 65.16.184.131	2018	-> 216.XXX.XXX.151	110	SYN	******S*
Mar	22	7:07:01 65.16.184.131	2019	-> 216.XXX.XXX.151	119	SYN	******S*
Mar	22	7:07:01 65.16.184.131	2020	-> 216.XXX.XXX.151	1080	SYN	******S*
Mar	22	7:07:01 65.16.184.131	2021	-> 216.XXX.XXX.151	6588	SYN	******S*
Mar	22	7:07:01 65.16.184.131	2022	-> 216.XXX.XXX.160	21	SYN	******S*
Mar	22	7:07:02 65.16.184.131	2024	-> 216.XXX.XXX.160	25	SYN	******S*
Mar	22	7:07:02 65.16.184.131	2025	-> 216.XXX.XXX.160	80	SYN	******S*
Mar	22	7:07:02 65.16.184.131	2026	-> 216.XXX.XXX.160	110	SYN	******S*
Mar	22	7:07:01 65.16.184.131	2027	-> 216.XXX.XXX.160	119	SYN	******S*
Mar	22	7:07:01 65.16.184.131	2028	-> 216.XXX.XXX.160	1080	SYN	******S*
Mar	22	7:07:01 65.16.184.131	2029	-> 216.XXX.XXX.160	6588	SYN	******S*
Mar	22	7:07:01 65.16.184.131	2030	-> 216.XXX.XXX.161	21	SYN	******S*
Mar	22	7:07:01 65.16.184.131	2032	-> 216.XXX.XXX.161	25	SYN	******S*
Mar	22	7:07:01 65.16.184.131	2033	-> 216.XXX.XXX.161	80	SYN	******S*
Mar	22	7:07:02 65.16.184.131	2034	-> 216.XXX.XXX.161	110	SYN	******S*
Mar	22	7:07:02 65.16.184.131	2035	-> 216.XXX.XXX.161	119	SYN	******S*
Mar	22	7:07:03 65.16.184.131	2036	-> 216.XXX.XXX.161	1080	SYN	******S*
Mar	22	7:07:03 65.16.184.131	2037	-> 216.XXX.XXX.161	6588	SYN	******S*
Mar	22	7:07:03 65.16.184.131	2038	-> 216.XXX.XXX.162	21	SYN	******S*
Mar	22	7:07:03 65.16.184.131	2040	-> 216.XXX.XXX.162	25	SYN	******S*
Mar	22	7:07:03 65.16.184.131	2041	-> 216.XXX.XXX.162	80	SYN	******S*
Mar	22	7:07:03 65.16.184.131	2042	-> 216.XXX.XXX.162	110	SYN	******S*
Mar	22	7:07:03 65.16.184.131	2043	-> 216.XXX.XXX.162	119	SYN	******S*
Mar	22	7:07:03 65.16.184.131	2044	-> 216.XXX.XXX.162	1080	SYN	******S*
Mar	22	7:07:03 65.16.184.131	2045	-> 216.XXX.XXX.162	6588	SYN	******S*

Fig 14-1

[**] [1:615:1] SCAN Proxy attempt [**]
[Classification: Attempted Information Leak] [Priority: 2]
03/24/02-02:18:35.743652 0:B0:64:B9:9E:C0 -> 0:50:4:1B:A6:51 type:0x800 len:0x3E
65.16.184.131:3524 -> 216.XXX.XXX.165:1080 TCP TTL:115 TOS:0x0 ID:64300 IpLen:20 DgmLen:48 DF
******S* Seq: 0xD4222E2E Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

[**] [1:615:1] SCAN Proxy attempt [**]
[Classification: Attempted Information Leak] [Priority: 2]

03/24/02-02:18:36.384437 0:B0:64:B9:9E:C0 -> 0:50:4:1B:A6:51 type:0x800 len:0x3E
65.16.184.131:3524 -> 216.XXX.XXX.165:1080 TCP TTL:115 TOS:0x0 ID:64393 IpLen:20 DgmLen:48 DF
******S* Seq: 0xD4222E2E Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

[**] [1:615:1] SCAN Proxy attempt [**]
[Classification: Attempted Information Leak] [Priority: 2]
03/24/02-02:18:37.044294 0:B0:64:B9:9E:C0 -> 0:50:4:1B:A6:51 type:0x800 len:0x3E
65.16.184.131:3524 -> 216.XXX.XXX.165:1080 TCP TTL:115 TOS:0x0 ID:64465 IpLen:20 DgmLen:48 DF
******S* Seq: 0xD4222E2E Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

[**] [1:615:1] SCAN Proxy attempt [**]
[Classification: Attempted Information Leak] [Priority: 2]
03/24/02-02:18:57.008034 0:B0:64:B9:9E:C0 -> 0:50:4:1B:A6:51 type:0x800 len:0x3E
65.16.184.131:1058 -> 216.XXX.XXX.164:1080 TCP TTL:115 TOS:0x0 ID:1303 IpLen:20 DgmLen:48 DF
******S* Seq: 0xD6C20718 Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

[**] [1:615:1] SCAN Proxy attempt [**]
[Classification: Attempted Information Leak] [Priority: 2]
03/24/02-02:18:57.605602 0:B0:64:B9:9E:C0 -> 0:50:4:1B:A6:51 type:0x800 len:0x3E
65.16.184.131:1058 -> 216.XXX.XXX.164:1080 TCP TTL:115 TOS:0x0 ID:1388 IpLen:20 DgmLen:48 DF
******S* Seq: 0xD6C20718 Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

[**] [1:615:1] SCAN Proxy attempt [**]
[Classification: Attempted Information Leak] [Priority: 2]
03/24/02-02:18:58.263712 0:B0:64:B9:9E:C0 -> 0:50:4:1B:A6:51 type:0x800 len:0x3E
65.16.184.131:1058 -> 216.XXX.XXX.164:1080 TCP TTL:115 TOS:0x0 ID:1478 IpLen:20 DgmLen:48 DF
******S* Seq: 0xD6C20718 Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

[**] [1:615:1] SCAN Proxy attempt [**]
[Classification: Attempted Information Leak] [Priority: 2]
03/24/02-02:19:18.313742 0:B0:64:B9:9E:C0 -> 0:50:4:1B:A6:51 type:0x800 len:0x3E
65.16.184.131:1912 -> 216.XXX.XXX.163:1080 TCP TTL:115 TOS:0x0 ID:3973 IpLen:20 DgmLen:48 DF
******S* Seq: 0xD9702793 Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

[**] [1:615:1] SCAN Proxy attempt [**]
[Classification: Attempted Information Leak] [Priority: 2]
03/24/02-02:19:18.929997 0:B0:64:B9:9E:C0 -> 0:50:4:1B:A6:51 type:0x800 len:0x3E
65.16.184.131:1912 -> 216.XXX.XXX.163:1080 TCP TTL:115 TOS:0x0 ID:4006 IpLen:20 DgmLen:48 DF
******S* Seq: 0xD9702793 Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

[**] [1:615:1] SCAN Proxy attempt [**]
[Classification: Attempted Information Leak] [Priority: 2]
03/24/02-02:19:19.480463 0:B0:64:B9:9E:C0 -> 0:50:4:1B:A6:51 type:0x800 len:0x3E
65.16.184.131:1912 -> 216.XXX.XXX.163:1080 TCP TTL:115 TOS:0x0 ID:4040 IpLen:20 DgmLen:48 DF
******S* Seq: 0xD9702793 Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK

Fig. 14-2

Source of Trace

The above alerts (**Fig 14-1 and 14-2)** were gathered using the Snort sensor in **Fig. 1-2.** The information was gathered on a FreeBSD 4.5-STABLE release system. Using the Unix command syntax in **Fig. 1**

How was the Detect Generated?

The traffic above was detected using Snort version 1.8.3 for Unix, and parsed from the logs using SnortSnarf. In addition, the information was sorted in Microsoft Excel. Within the snort rule set, there were no matching rules to gather the additional data at that time. The scan was picked up as a Proxy Scan due to the probing of port 1080.

The rules that gathered this information are:
preprocessor portscan: $HOME_NET 4 3 portscan.log

alert tcp $EXTERNAL_NET any -> $HOME_NET 1080 (msg:"SCAN Proxy attempt";flags:S; classtype:attempted-recon; sid:615; rev:1;)

Probability the Source Address was Spoofed

As an Intrusion Analyst the question in regards to the traffic being spoofed must be factored in. What are the chances the source address was spoofed? The traffic identified in **Fig 14-1** and **Fig. 14-2** is very likely not spoofed. This is determined by looking at the source ports and how they are sequenced, indicating the attacking was actually making the connection or doing recon for active connections. In addition, if you look at the dates it appears the attacker decided to return and attempts to scan the network again for an open proxy port.

Description of Attack

This attack appears to target standard proxy servers. The attack that took place on the 22nd of March 2002 appears to have been searching for an 'Analog X' proxy system. (Before continuing, a Special Thanks to Johannes B. Ullrich, as he assisted in the identification of this scan). This scan is identified by the targeting of ports 21,25,80,110,119,1080, 6588. It appears the attacker came back to verify results found on the network since they scanned for proxies on the network as well. The attacker could have been searching for open mail relay. It is very difficult to determine the attacker's state of mind though we can be assured they were searching for a system to exploit.

Attack Mechanism

Was this attack a Stimulus or Response?
This attack would be classified as stimulus. The stimulus is indicated by the SYN packets. A SYN/ACK response would have indicated a response to the connection traffic. In addition, the system that was attacked does have some of those ports open. If there was a response we would have seen a SYN/ACK arriving to a port >1024. Since normal TCP communication traffic occurs at ports >1024 and those ports <1024 are reserved in most cases for services.

Questions asked to determine Stimulus or Response

Are there any services that communicate on this port, which would have drawn this type of traffic?
Yes/No, some of the ports contain valid services such as with 21,25,80,110. The 119,1080,6588 do not contain any services on the system. Due to the presence of some valid services a evaluation is necessary of the system to ensure

What service(s) is/are being targeted?
Upon review of the snort alerts it appears the attacker was targeting port 1080. However, the scan of the other ports was perhaps to locate a specific application called "Analog X Proxy" server. When the attacker returned 2 days later, they specifically targeted only port 1080, which is not in service on this system.
In addition, the attacker could have been seeking a Open Mail relay and was attempting to throw off anyone monitoring by scanning the other ports. Though I doubt this since the attacker did come back as mentioned earlier and target port 1080.

Does the service have known vulnerabilities?
Recently, there have been vulnerabilities released, that exploit the AnalogX application as well as Squid Proxy. This traffic can be a reconnaissance searching for affected systems.

Are there attack tools present for these services?
Yes, there are vulnerabilities for some versions of proxy services as well for the AnalogX application.

With all these conditions present it would be safe to say that this was a stimulus and not a response to some traffic.

Correlation

This attack arrived in correlation to some events that were posted on the DSHEILD.ORG message board for March 2002 by Clay Dillard. In addition, this was confirmed by

checking the DSHEILD.ORG website and confirmed there was an increase in scans against the ports 8080 and 1080. This possibly could be the result of a new upcoming vulnerability, or the resurrection of an old one. In addition, proxy services allow the attacker to remain anonymous, so this could be the result of an attacker attempting to further hide or attempt to hide their identity.

Evidence of Active Targeting

Is there evidence of Active Targeting with this alert? In order to answer this question we must first investigate the following questions and determine their results.

Is the activity targeting a specific host?
No, this activity scanned the monitored network range seeking information, and seeking responses to certain services.
Is this a general scan of entire network?
Yes, this scan appears to be a reconnaissance of the entire monitored network. The activity indicates that this was a recon scan seeking a specific application or ports of value to possibly exploit.

Is this a probable "wrong number"?
No, the determination on this comes from the fact the attacker returned 2 days later to follow up on the results. They did this by accomplishing an additional Proxy scan on the network possibly to confirm their results.

With those questions answered, the findings indicate the initial incident would not have been active targeting though the follow up activity accomplished by this intruder would be classified as active targeting.

Severity of Attack

(Criticality + lethality) – (system + network countermeasures) = severity

Criticality = 5
The system being attacked has some of the ports that were scanned open. This poses at least some concern. If this system were to be compromised, it would represent a significant problem with customer communication, since this system is used as web, mail, pop3, and other services on the same server. This poses a significant risk if the system were to be compromised by an attacker. In addition, since there is the possibility the attacker is gathering information for a remote root exploit, as is the case with the AnalogX exploit and Squid Exploits this makes this critical. This could possibly allow remote root access to the system, if the attack ended up being successful.

Lethality = 2
The attack as indicated was a scan so its lethality is low.

System Countermeasures = 4
The system is patched with all necessary patches up to date. In addition, a majority of the services does not run as root, they are run either as "standard users" or as their own users without root privileges. In addition, there is no proxy service running on this system, which appears to be what the attacker was targeting. This host also contains a host-based IDS/Firewall using IP tables to protect it from malicious users.

Network Countermeasures = 1
This system has IPFW installed but we are unable to block a majority of the traffic due to the legitimate services that are running on the ports and the access from outside users.

(Criticality + Lethality) – (System + Network Countermeasures) = Severity
(5 + 2) – (4 + 1) = 2

With a severity level of 2 this system should be checked and verify no intrusion has taken place.

Protection Measures

In order to tighten security for this system some of the services should be closed. Additionally, some of the services should be moved to additional systems. This could possibly lower the criticality of the system, so that a single compromise does not have the potential of wiping out a wide array of crucial systems/services. Furthermore, installing a host-based IDS such as tripwire may allow you to quickly identify if unauthorized changes to files have taken place and adds an additional layer of protection to the targeted system.

Test Question #3

A _____ IDS could protect file changes on the individual system as well as, can assist in blocking malicious traffic destined for the host.

A. Network-Based
B. Firewall
C. Router
D. Host-Based

Answer:
D Host-Based

A host-based IDS system will protect files on this individual host as well as has the capability to protect the targeted system from attacks at the system. A Network-based IDS allows us to identify traffic and provides some measure of protection but CANNOT offer the security of the files on the system. A firewall will block from external sources in most cases but, what about the internal sources that we may need to protect from? Same situation with a router we could use ACL (Access control lists) to protect our systems from traffic arriving through the router but it cannot protect files and if the systems are

one the same switch, we are not protected. Therefore, the only real way to protect the system files would be to use host-based IDS in this situation.

Alert #4

FTP Port 21/Anonymous FTP / Warez Dump Attempt

Snort Detect

[**] [1:0:0] Connection to HoneyPot FTP [**]
03/19/02-14:50:11.034576 0:10:67:0:4E:5B -> 0:0:39:61:50:2E type:0x800 len:0x3E
172.180.48.64:2002 -> 198.xxx.xxx.209:21 TCP TTL:112 TOS:0x0 ID:20375 IpLen:20 DgmLen:48 DF
******S* Seq: 0xC7A06F58 Ack: 0x0 Win: 0x2238 TcpLen: 28
TCP Options (4) => MSS: 1360 NOP NOP SackOK

[**] [1:0:0] Connection to HoneyPot FTP [**]
03/19/02-14:50:11.034660 0:0:39:61:50:2E -> 0:10:67:0:4E:5B type:0x800 len:0x3A
198.xxx.xxx.209:21 -> 172.180.48.64:2002 TCP TTL:128 TOS:0x0 ID:23910 IpLen:20 DgmLen:44 DF
***A**S* Seq: 0x47C60B85 Ack: 0xC7A06F59 Win: 0x2530 TcpLen: 24
TCP Options (1) => MSS: 1460

[**] [1:0:0] Connection to HoneyPot FTP [**]
03/19/02-14:50:13.479290 0:10:67:0:4E:5B -> 0:0:39:61:50:2E type:0x800 len:0x3C
172.180.48.64:2002 -> 198.xxx.xxx.209:21 TCP TTL:112 TOS:0x0 ID:20712 IpLen:20 DgmLen:40 DF
A* Seq: 0xC7A06F59 Ack: 0x47C60BC5 Win: 0x24F1 TcpLen: 20

[**] [1:553:2] INFO FTP anonymous login attempt [**]
[Classification: Misc activity] [Priority: 3]
03/19/02-14:50:14.517187 0:10:67:0:4E:5B -> 0:0:39:61:50:2E type:0x800 len:0x46
172.180.48.64:2002 -> 198.xxx.xxx.209:21 TCP TTL:112 TOS:0x0 ID:20847 IpLen:20 DgmLen:56 DF
AP Seq: 0xC7A06F59 Ack: 0x47C60BC7 Win: 0x24EF TcpLen: 20

Fig. 21-1

Specter IDS Detection/HoneyPot

```
************************************************************
  System name : OUTPOST
  Config file version : 1.0
  Maximum connections : 5
  Connection throttle : on
  Connections/min. : 10
  Flood blocking : off
  Send status mail : no
```

```
  Send mails : no
  Send short mails : no
  Log to files : yes
  Log to event log : no
  Log to syslog : no
  Do finger probe : no
  Do port scan : no
  Whois lookup : yes
  Log telnet banner : no
  Log ftp banner : no
  Log smtp banner : no
  Log http document : no
  Log http header : no
  Custom warning msg. : yes
  Custom POP3 msg. : no
  Provide POP3 msg. : yes
  Use web graphics : no
  Use custom web doc. : no
  Expect friendly con. : no
  Remote management : no
  Trace route : yes
  Maximum hops : 30
  Do reverse lookup : yes
  Send password files : yes
  Password type : easy
  Activated services : FTP TELNET SMTP POP3 NETBUS FINGER HTTP
  Activated traps : DNS SUN-RPC SUBSEVEN SSH IMAP BO2K UPNP
  Generic trap port : 5000
  Mail Server :
  Mail Address :
  Short Mail Address :
  Role OS : Windows NT
  Role Character : Open System
  Role Hostname : system.britsys.net
  Crowd Level : Multiple users
  User Names : Default + Custom
************************************************************

Trace route information :
Tracing route to 172.180.48.64 with 32 bytes of data:

1  (100ms)   198.XXX.XXX.225
2  (100ms)   207.112.240.201  (e3-11.nchicago2-core0.bbnplanet.net)
3  (80ms)    4.0.3.125        (p3-0.nchicago2-cr2.bbnplanet.net)
4  (90ms)    4.0.5.241        (p7-3.chcgil2-cr9.bbnplanet.net)
5  (90ms)    4.24.8.109       (so-3-2-0.chcgil2-br1.bbnplanet.net)
6  (91ms)    4.24.5.218       (so-7-0-0.chcgil2-br2.bbnplanet.net)
7  (90ms)    4.24.9.34        (p1-0.chcgil2-cr2.bbnplanet.net)
8  (90ms)    4.24.203.30      (a0-0.xchcgil4-uunet.bbnplanet.net)
9  (90ms)    152.63.68.6      (0.so-5-2-0.XL2.CHI2.ALTER.NET)
10 (90ms)    152.63.67.121    (0.so-1-0-0.TL2.CHI2.ALTER.NET)
11 (110ms)   152.63.19.170    (0.so-3-0-0.TL2.DCA6.ALTER.NET)
12 (110ms)   152.63.38.33     (0.so-6-0-0.XL2.IAD1.ALTER.NET)
13 (110ms)   152.63.6.201     (POS6-0.GW1.IAD1.ALTER.NET)
14 (110ms)   66.185.140.137   (pop3-rtc-P6-0.atdn.net)
15 (110ms)   66.185.140.129   (bb1-rtc-P14-0.atdn.net)
16 (110ms)   66.185.153.1     (bb1-dcl-P5-0.atdn.net)
```

```
17 (110ms)   66.185.153.170   (bb1-nyc-P4-0.atdn.net)
18 (210ms)   66.185.152.65    (bb1-loh-P3-0.atdn.net)
19 (210ms)   66.185.146.66    (pop4-loh-P0-0.atdn.net)
20 (210ms)   66.185.146.74    (accessl1-loh-P0-0.atdn.net)
21 (210ms)   195.93.52.46     (supportl4-loh-P0-0.router.aol.com)
22 (210ms)   195.93.49.228    (rt-loh49.proxy.aol.com)
23 (631ms)   172.180.48.64    (ACB43040.ipt.aol.com)

Remote host is 23 hops away.

************************************************************

Whois information :
   12100 Sunrise Valley Drive
   Reston, VA 20191
   US
   Netname: AOL-172BLK
   Netblock: 172.128.0.0 - 172.191.255.255
   Maintainer: AOL

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

   Domain System inverse mapping provided by:
   DAHA-01.NS.AOL.COM         152.163.159.233
   DAHA-02.NS.AOL.COM         205.188.157.233
   ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE

   Record last updated on 28-Mar-2001.
   Database last updated on  18-Mar-2002 19:58:22 EDT.

The ARIN Registration Services Host contains ONLY Internet
Network Information: Networks, ASN's, and related POC's.
Please use the whois server at rs.internic.net for DOMAIN related
Information and whois.nic.mil for NIPRNET Information.
************************************************************
FTP connection
Host  : 172.180.48.64  (ACB43040.ipt.aol.com)
Login : anonymous
Pass  : Hgpuser@home.com
Time  : Tue Mar 19 06:50:11 2002

Log :
Client connecting: 172.180.48.64
Client tries anonymous Login
--->331 Anonymous login ok, please send your e-mail address as
password.
Client sent PASS 'Hgpuser@home.com'
--->230 User anonymous logged in.
Client wants to change current directory to
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to _vti_pvt/
--->200 CWD command successful.
Client wants to create directory
```

```
--->550 Permission denied.
Client wants to change current directory to _vti_txt/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to _vti_cfg/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to _vti_log/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to _vti_cnf/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to _vti_bin/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to _vti_usr/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to _vti_tmp/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to _vti_temp/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to _vti_html/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to _vti_images/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to _private/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to incoming/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to outgoing/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to public/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
```

```
Client wants to change current directory to public/incoming/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to public/outgoing/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to public_html/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/incoming/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/outgoing/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/images/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/_vti_pvt/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/_vti_txt/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/_vti_cfg/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/_vti_log/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/_vti_cnf/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/_vti_bin/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/_vti_usr/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/_vti_tmp/
```

```
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/_vti_temp/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/_vti_html/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to pub/_vti_images/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to upload/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to wwwroot/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to wwwroot/pub/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to wwwroot/public/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to wwwroot/incoming/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to wwwroot/outgoing/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to mailroot/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to ftproot/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to home/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to images/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to web/
--->200 CWD command successful.
```

```
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to www/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to html/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to cgi-bin/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to usr/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to usr/incoming/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to usr/outgoing/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to temp/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to ~temp/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to tmp/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to ~tmp/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/pub/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/public/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/incoming/
--->200 CWD command successful.
Client wants to create directory
```

```
--->550 Permission denied.
Client wants to change current directory to anonymous/outgoing/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/_vti_pvt/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/_vti_cnf/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/_vti_log/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/_vti_txt/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/_vti_cfg/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/_vti_bin/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/_vti_usr/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/_vti_tmp/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/_vti_temp/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/_vti_html/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Client wants to change current directory to anonymous/_vti_images/
--->200 CWD command successful.
Client wants to create directory
--->550 Permission denied.
Connection timed out
```

Fig 21-2

Source of Trace

The detects from (**Fig 21-1 and 21-2)** were gathered using the configuration in **Fig 1-1**. The attack was directed at the Specter IDS/Snort sensor in **Fig. 1-1.** The system hosting the Specter IDS/Snort system is a Windows NT 4.0 Server with SP6a and all applicable security patches are installed.

How was the Detect Generated?

The traffic above was detected using Snort version 1.8.3 for Win32, and arranged and parsed with SnortSnarf on a FreeBSD Unix system. In addition, the presence of the honeypot has allowed us to see what the attacker was attempting to accomplish. Within the snort rule set, there were no matching rules to gather the additional data directed at the FTP so I wrote my own rule that would track all information directed at the Honeypot FTP. See **Fig 29-1** for the rules that were used to gather this traffic. The traffic identified by Snort has been truncated for easier viewing and identification. The Specter IDS system traffic shows the entire traffic logs from this host.

The rules that gathered this information are:

alert tcp $EXTERNAL_NET any <> $HONEY_NET 21 (msg:"Connection to Honeypot FTP";flags:S+;)

alert tcp $EXTERNAL_NET any -> $HOME_NET 21 (msg:"INFO FTP anonymous login attempt"; content:"USER anonymous|0D0A|"; nocase; flags:A+; classtype:misc-activity; sid:553; rev:2;)

Fig 29-1

Probability the Source Address was Spoofed

In order to determine if this traffic is actually from the offending host there is the issue of spoofing that must be taken into consideration. We can be assured there is no spoofing involved in this incident since the attacker has completed the three-way handshake. (SYN – SYN/ACK- ACK). This can be viewed in **Fig 21-1.**

Description of Attack

This attack has targeted FTP servers where anonymous FTP is enabled. In addition this attack seeks to find directories where it can create and store files. This is proven by the activities in **Fig 21-2**. In this traffic the attacker is attempting to locate any directory where they have the ability to create additional directories or place files on the server.

Attack Mechanism

Was this attack a Stimulus or Response?
This attack contains both stimulus and response properties, due to the honeypot involvement in the communication. The initial packet in **Fig 21-1,** is considered the stimulus, and the subsequent SYN/ACK in return from the honeypot is considered the response. Since the honeypot did not initiate the connection this traffic overall, the initial attack is classified as the stimuli.

Questions asked to determine Stimulus or Response

Are there any services that communicate on this port, which would have drawn this type of traffic?
Yes, the FTP service is open on the targeted system. This port was opened due to the high volume of FTP scans that had been received. This was open in order to identify what the attackers really were attempting to download/upload onto the system.

What service(s) is/are being targeted?
The service that was being targeted was the FTP service. The traffic in **Fig 21-1 and 21-2** was the only traffic directed at the system from this host.

Does the service have known vulnerabilities?
No, after a search on the internet for possible exploits for this software there did not appear to be any available. This does not mean there is no exploits available due to the concern that a hacker can keep the exploit for only limited usage and not release to many entities in the worry it will fall into the hands of security experts. The attacker did attempt to use a common misconfiguration of the server. This misconfiguration would be the ability to get into the system using the anonymous FTP access.

Are there attack tools present for these services?
No, there were no attack tools present except for the fact the attacker logged in anonymously and attempted to locate directories where they can place or download software or for other uses.

With all these conditions present it would be safe to say that this was a stimulus and not a response to some traffic.

Correlation
There are so many attempts at FTP probes and connections. In order to correlate this to some activity all that is needed is for one to check on the lists at DSHEILD.ORG. In addition, there are so many probes this traffic can be expected due to its prevalence on the internet. Moreover there are so many systems that are improperly setup and configured this causes many of the probes on the internet.

Evidence of Active Targeting

Is there evidence of Active Targeting with this alert? In order to effectively answer the question in regards to this, an determination of the following questions must be performed.

Is the activity targeting a specific host?
Yes, this activity targeted this system and completed the connection to this system. There was no scan to correlate to other systems on the network. In addition the attacker specifically connected to the system and attempted to create directories on the system.

Is this a general scan of entire network?
No, this scan was directed at the specific system and was not a reconnaissance attack. In addition, the attacker connected to the system and targeted the single system on this network.

Is this a probable "wrong number"?
A "wrong number" scenario is possible though highly unlikely. While the possibility still exists, that this attack could be a wrong number, the attacker would have to know the internet address of the system since the DNS name lookup does not work on this system. Additionally, the attacker attempted to create directories in folders that would only exist on Windows Systems as well as possible Unix systems; this indication shows that the attacker never bothered to recon the system.

With those questions answered, the findings indicate that some active targeting has occurred in this system.

Severity of Attack

(Criticality + lethality) – (system + network countermeasures) = severity

Criticality = 1
The criticality on this system is very low since it is a honeypot system. In addition, the system does not have any true services running on the system. All patches have been installed and are current.

Lethality = 1
The lethality of this attack appears to be very low. The attacker appears to be probing for a directory that is writeable. This could be so that the attacker can copy "warez" (pirated software), viruses, or other malicious code to a location. The complete effect of this is very low as far as the lethality of the attack is concerned.

System Countermeasures = 4
The system has the honeypot on it, as well as there are no true services located on the system. All security patches have been installed on the system. This can be seen below in **Fig 31-1**.

SuperScan on Localhost

NO HONEYPOT ACTIVE

* + 127.0.0.1

|___ 135 DCE endpoint resolution

HONEYPOT ACTIVE

*** + 127.0.0.1**

|___ 22 SSH Remote Login Protocol

|___ 23 Telnet

|___ ...

|___ 25 Simple Mail Transfer

|___ 220 system.myhome.net ESMTP Sendmail 8.8.8/8.8.8; Mon Apr 01 07:50:56 2002.

|___ 53 Domain Name Server

|___ 79 Finger

|___ Login Name Tty Idle Login Time Office Office Phone

|___ 110 Post Office Protocol - Version 3

|___ +OK QPOP (version 2.53) at system.myhome.net starting.

|___ 111 SUN Remote Procedure Call

|___ 135 DCE endpoint resolution

|___ 143 Internet Message Access Protocol

|___ 515 spooler

Fig 31-1

Network Countermeasures = 2

The network currently does not provide much protection but does have the presence of a Network Based IDS. This assists in identifying malicious traffic directed at this network.

(Criticality + lethality) – (system + network countermeasures) = severity

(1 + 1) – (4 + 2) = -4

With a rating of -4 our severity of this attack is negligible. This attack could not succeed due to the presence of the honeypot program. The honeypot program, allows the attacker to see a false OS as well as FTP, in addition the honeypot tracks all activity of the attacker.

Protection Measures

The protection measures that are recommended in order to protect against this attack would be to disable anonymous ftp unless specifically needed. In addition, the analyst should ensure that the FTP server patches have been applied as well as OS patches. The system should be placed as a bastion host so that the internal network is protected, in case of the system being compromised. If the system is for internal usage the system should be placed inside the firewall and have access restricted based upon network or ip address as determined by business and security necessities.

Test Question #3

A _____ is a system that is designed to simulate one or more network services on a computer system.

a) Network-based IDS
b) Host-based IDS
c) Honeypot
d) Router

Answer:
C
A honeypot is a system designed to simulate one or more network services. This type of system allows an attacker to think they have accomplished the goal of breaking into the system while the system is still secured. It falsifies responses and communicates with the attacker. During the attack, process the honeypot will log all responses and actions by the attacker as well as the "server". A host-based IDS provides the additional features that can complement the honeypot's effectiveness.

Alert #5

Possible CDE Buffer Overflow Attempt

Mar 31 02:01:59 194.206.91.3:4484 -> 216.XXX.XXX.150:6112 SYN ******S*
Mar 31 02:01:59 194.206.91.3:4485 -> 216.XXX.XXX.151:6112 SYN ******S*
Mar 31 02:01:59 194.206.91.3:4494 -> 216.XXX.XXX.160:6112 SYN ******S*
Mar 31 02:01:59 194.206.91.3:4495 -> 216.XXX.XXX.161:6112 SYN ******S*
Mar 31 02:01:59 194.206.91.3:4497 -> 216.XXX.XXX.163:6112 SYN ******S*
Mar 31 02:01:59 194.206.91.3:4496 -> 216.XXX.XXX.162:6112 SYN ******S*
Mar 31 02:01:59 194.206.91.3:4498 -> 216.XXX.XXX.164:6112 SYN ******S*
Mar 31 02:01:59 194.206.91.3:4499 -> 216.XXX.XXX.165:6112 SYN ******S*

Fig. 34-1

Mar 30 06:07:09 202.102.29.102:48667 -> 216.XXX.XXX.150:6112 SYN ******S*
Mar 30 06:07:09 202.102.29.102:48668 -> 216.XXX.XXX.151:6112 SYN ******S*
Mar 30 06:07:09 202.102.29.102:48677 -> 216.XXX.XXX.160:6112 SYN ******S*

Mar 30 06:07:09 202.102.29.102:48682 -> 216.XXX.XXX.165:6112 SYN ******S*
Mar 30 06:07:09 202.102.29.102:48678 -> 216.XXX.XXX.161:6112 SYN ******S*
Mar 30 06:07:09 202.102.29.102:48680 -> 216.XXX.XXX.163:6112 SYN ******S*
Mar 30 06:07:09 202.102.29.102:48681 -> 216.XXX.XXX.164:6112 SYN ******S*
Mar 30 06:07:09 202.102.29.102:48679 -> 216.XXX.XXX.162:6112 SYN ******S*

Fig. 34-2

Source of Trace

The portscan alerts located above, were pulled from logs located on the Snort Sensor in **Fig. 1.2.** They were parsed by the SnortSnarf application.

How was the Detect Generated?

The traffic identified in **Fig. 34-1** and **Fig. 34-2,** were gathered using Snort version 1.8.3 for FreeBSD 4.5-Stable and parsed with SnortSnarf on the same FreeBSD Unix system. The rule used to gather this information is located below in **Fig. 35-1.**

preprocessor portscan: $HOME_NET 4 3 portscan.log
Fig. 35-1

Probability the Source Address was Spoofed

The probability that the source address was spoofed is low. This is determined by the fact that TCP was used, which requires the three-way handshake to be completed in order to successfully complete the connection. If the attacker was spoofing the connection, they would not receive back the SYN/ACK response from the server but the "spoofed" host would receive that information. In addition, the close sequencing of the port numbers in both cases from **Fig. 34-1** and **Fig. 34-2** indicate that the attacking system is doing the scanning and not a script to falsify information.

Description of Attack

This attack could be targeting a potential vulnerability in CDE that exists on Sun systems. This was identified by tracking the service, which is known to run on port tcp/6112 and udp/6112. Unfortunately, with the way the Snort sensor is currently setup, the ShellCode rules were turned off. This handcuffed our ability to determine if a ShellCode exploit was run prior which is a true signature of the buffer overflow attempt.

Attack Mechanism

Was this attack a Stimulus or Response?

This attack can be classified as stimuli since the server did not initiate the communication process. In addition, the server would have responded with SYN/ACK as the response if it were to be classified as response in this case. The attacker sent a SYN packet to the systems on the network, which classifies this as a stimulus.

What service(s) is/are being targeted?

The service being targeted by this attack is CDE Subprocess Control Service, or dtspcd located on most systems running CDE as well as by default on most Sun systems.

Does the service have known vulnerabilities?

Yes, the service does have known vulnerabilities. The advisory in regards to this vulnerability was released November 12, 2001 by the Computer Emergency Response Team (CERT).
CERT Advisory CA-2001-31 Buffer Overflow in CDE Subprocess Control Service
CERT Advisory CA-2002-01 Exploitation of Vulnerability in CDE Subprocess Control Service
CERT Vulnerability Note # 172583: Common Desktop Environment Subprocess Control Service dtspcd contains buffer overflow

In addition, there is some further information located on the Counterpane Internet Security website at the following URL: http://www.counterpane.com/alert-cde.html

Are there attack tools present for these services?

Currently there are attack tools being used in the wild for this attack. This information was gathered from SecurityFocus. According to their exploit page, they have indicated that this exploit is in the wild.

With these conditions, the indication is that the attack was the stimulus hoping to find a response from a system running and using CDE so that a potentially successful could occur.

Correlation

This traffic can be correlated with additional traffic and due to the presence of exploit in the wild; most assured, there are script-kiddies or other attacker out there looking to exploit the administrator who did not patch their systems. From the amount of traffic received lately on the Snort Sensor as well as the honeypot from **Fig. 1-1,** it appears scans to gather information have increased while direct assaults have been steady.

Evidence of Active Targeting

Is the activity targeting a specific host?
No, this activity was directed at the network. According to the patterns derived from the Snort Portscan.log it appears the attacker was scanning the network attempting to locate a potential target for the CDE Buffer Overflow attack.

Is this a general scan of entire network?
Yes, this attack appears to be a scan of the network directed at port tcp port 6112. There was the SYN connection but no further communication from this host in the following or proceeding days. In addition, according to the traffic located in **Fig. 34-1** and **Fig. 34-2** the monitored internet addresses were attacked sequentially.

Is this a probable "wrong number"?
No, this scan does not have the indications of a wrong number. A wrong number would have some characteristics such as random ip addresses or ports or perhaps a few systems targeted. This appears more like a reconnaissance attack, being used to gather evidence or potential victims.

From the evidence above there does not appear to be active targeting. There does appear to be evidence of reconnaissance information gathering occurring and potential information leakage if the attacker locates a system to attack.

Severity of Attack

(Criticality + lethality) – (system + network countermeasures) = severity

Criticality = 5
The criticality of this attack is very high due to the services that are located on this system. This system contains DNS and Mail services, which are in turn, classified as critical services.

Lethality = 1
The lethality of this attack is low since this is a probe for a service and we are unable to determine if it was a legitimate attack. The lethality of this attack would be much higher if the service was running on the system.

System Countermeasures = 5
The system does not have CDE running and does not have any services that are bound to port 6112. This causes the attacker to not receive any information. Furthermore, the system is a FreeBSD system, which does not use CDE.

Network Countermeasures = 3
The countermeasures present include a log checker, as well Snort, which will log and identify any malicious traffic. In addition, since there was no SYN/ACK traffic present

there was no connection accomplished. Furthermore, there is a host-based firewall located on this system, which will filter traffic directed at ports other than the specified ports.

(Criticality + lethality) – (system + network countermeasures) = severity
(5 + 1) – (5 + 3) = -2

With a severity of –2 there is very little concern for the attack. This is especially true since we know this is not a Sun System or other possible system which could be attacked with this exploit.

Protection Measures

Some protection measures that could be implemented would be to block all ports on the system that we are sure will not be used and open them as necessary. This will protect the host. In addition, in order to protect the system at the network layer we could implement the ShellCode rules in Snort so that we could positively identify if this attack was truly a Buffer Overflow probe or merely a scan seeking the port to attempt later. Without the Shell Code rules we may never know if this attack was a true attack or merely a scan. The fixes should be implemented in order to detect for this attack.

Question #5

Which rule below would **increase** the time interval between port connections for detecting portscans and triggering an alert?
(Default is **preprocessor portscan: $HOME_NET 4 3 portscan.log)**

a) preprocessor portscan: $HOME_NET 3 3 portscan.log
b) preprocessor portscan: $HOME_NET 3 2 portscan.log
c) preprocessor portscan: $HOME_NET 4 4 portscan.log
d) preprocessor portscan: $HOME_NET 5 2 portscan.log

Answer: C
The answer is C.
Preprocessor tells snort to Process this module before the detection engine is "engaged". Portscan identifies the plugin. $HOME_NET identifies the traffic from which to look (identified in your Snort.conf file). The first number identifies the amount of ports that must be connected to and the second number identifies the amount of time in order to trigger this alert. The portscan.log is the file to write the alerts.

Assignment #3

Executive Overview

The network that the alerts were derived from appears to be very insecure from an initial glance. In addition, there are solid indications within the IDS alerts that various systems on the network are possibly compromised and need to have immediate attention directed to them. In order to accurately identify positive traffic from the false alerts there is a necessity to work on the rule files and remove what is unnecessary and clean up the environment. This will drastically improve the quality of the IDS capability as well as the Intrusion Analyst will not consistently be chasing false positives, and miss true positive events. Moreover, while looking through the directory located at http://www.research.umbc.edu/~andy it appears the traffic overall for the network is rapidly increasing as well as alerts on this network is increasing. This tends to raise red flags, or at least it does from the current perspective.

List of Files Used

alert.020324.gz	25-Mar-2002 00:05	1021k
scans.020324.gz	25-Mar-2002 00:11	1.6M
oos_Mar.24.2002.gz	25-Mar-2002 05:59	1k
alert.020325.gz	26-Mar-2002 00:05	1.8M
scans.020325.gz	26-Mar-2002 00:11	2.5M
oos_Mar.25.2002.gz	26-Mar-2002 06:01	1k
alert.020326.gz	27-Mar-2002 00:05	1.7M
scans.020326.gz	27-Mar-2002 00:11	2.3M
oos_Mar.26.2002.gz	27-Mar-2002 06:02	1k
alert.020327.gz	28-Mar-2002 00:06	1.7M
scans.020327.gz	28-Mar-2002 00:11	2.5M
oos_Mar.27.2002.gz	28-Mar-2002 06:04	1k
alert.020328.gz	29-Mar-2002 00:05	1.6M
scans.020328.gz	29-Mar-2002 00:11	2.2M
oos_Mar.28.2002.gz	29-Mar-2002 06:05	2k

Analysis Process

Due to the sheer size of the files listed above many programs could not parse the files and caused errors. In addition, the MY.NET substitution caused issues with many of the parsing programs that are available to parse the logs. So in order to get past these issues The files were FTP/SCP to a Unix system, and then manipulated so that MY.NET was changed to MY.NET.xxx.yyy. This allowed SnortSnarf to be able to parse the Alert files and allowed many of the scripts to better handle the data. The amount of data that was parsed was over 260,000 alerts that were identified during this timeframe.

The command used to change the files was:
Command:
for sfile in `ls alerts-02032*`; do cat $sfile | sed 's/MY.NET/MY.NET/g' >mb$sfile; done

For the scan files the same process was used except on a Windows system with ActiveState Perl installed. I used separate Operating systems to verify the integrity of the files and ensure the same results were reached on each system.
Once the files were parsed using SnortSnarf the files were manageable, and able to be parsed separately or together.
All files were concatenated together to make for more efficient processing and easier analysis.

Relationship on the Network

From the traffic on the network, it appears there is a mix of both Windows and Unix systems. The traffic that was gathered is based upon the volume of alerts generated by the traffic and plan on sorting it according to severity.

Top 10 Alerts

Priority	Signature	# Alerts	# Sources	# Dests
N/A	SMB Name Wildcard	54242	139	122
N/A	connect to 515 from inside	46587	63	4
N/A	spp_http_decode: IIS Unicode attack detected	44289	79	430
N/A	SNMP public access	40044	24	147
N/A	ICMP Echo Request L3retriever Ping	26828	92	13
N/A	MISC Large UDP Packet	21982	16	8
N/A	INFO MSN IM Chat data	5767	77	76
N/A	ICMP Echo Request Nmap or HPING2	3755	62	122
N/A	INFO Inbound GNUTella Connect request	2819	2338	7
N/A	ICMP Fragment Reassembly Time Exceeded	2105	25	53

This traffic identifies the top 10 alert generating rules within the rule set. Some of the rules appear to be made by the administrator and generating possibly unnecessary alerts. This can be determined from the "second" alert in the listing above. I will go into more detail in regards to this later in the assignment.

SMB Name Wildcard

Out of the 16,000 alerts that were generated for this traffic the majority arrived from a single host as indicated in **Fig. 3-1.** The traffic that generated this alert is caused by connections with a source and destination port of 137. Check the alert traffic below that notates the traffic that generated this alert. While this traffic could be legitimate, there is a cause for concern as well. As was noted in Bryce Alexander's practical and noted by the FAQ IDS questions of the same author it notes notable concerns regarding this traffic and its implementations. This traffic could signify a script-kiddy searching for systems to

exploit. Bryce Alexander's practical makes note of external IP addresses in correlation with the internal traffic such as in **Fig. 3-2**. This traffic is still a cause for concern and should be investigated and ensure that the source host does not have a virus or other Trojan installed that is allowing malicious traffic to be sent from it.
The system that is affected (the source system) should be thoroughly checked out and ensure it is not compromised. If the system is deemed clean and the traffic is legitimate, perhaps this rule should be removed or refined to allow better monitoring for the offending traffic.

```
03/24-00:01:12.032153 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.213:137
03/24-00:01:35.492258 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.20:137
03/24-00:02:07.304215 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.18:137
03/24-00:02:46.803013 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.21:137
03/24-00:02:47.878803 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.174:137
03/24-00:03:10.795280 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.13:137
03/24-00:03:16.082454 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.46:137
03/24-00:03:17.954567 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.172:137
03/24-00:03:41.805461 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.158:137
03/24-00:05:21.967240 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.10:137
03/24-00:06:00.209478 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.180:137
03/24-00:06:20.792513 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.178:137
03/24-00:07:05.250926 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.17:137
03/24-00:07:52.329561 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.177:137
03/24-00:07:56.444310 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.175:137
03/24-00:08:02.178072 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.44:137
03/24-00:08:42.459074 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.249:137
03/24-00:09:02.839258 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.167:137
```

Fig. 3-1

Apr 21 00:17:29 myhost snort: SMB Name Wildcard: MY.NET.0.1:137 -> my.ip.addr:137
Apr 21 00:17:29 myhost snort: SMB Name Wildcard: 24.28.135.131:137 -> my.ip.addr:137

Apr 21 00:17:31 myhost snort: SMB Name Wildcard: 24.28.135.131:137 -> my.ip.addr:137
Apr 21 00:17:31 myhost snort: SMB Name Wildcard: MY.NET.0.1:137 -> my.ip.addr:137

Fig. 3-2

Connect to 515 from inside

This alert was gathered by a rule that was manually written. Upon searching the snort rule sets there is no standard rule within Snort to capture this type of traffic. Additionally, it appears the traffic is communications with a printer on the network. The LPR service is what uses this port for communications. While there are some attacks, for the LPR services there are only 4 destination hosts on the network and a majority of the MY.NET hosts are communicating with these systems.

Top Scanners

The amount of Scans was staggering. I used a script found on the Internet and included in the Appendix to parse and gather the information of these scans.
The files for this were concatenated together to make for easier analysis and parsing.
The portscans were enumerated, sorted according to percentage, and volume/number of alerts.

The command to concatenate the files in this case was:

cat mbscans-02032* >combinedscans.txt

This combined all the files into the single file combinedscans.txt
The size on this file was approximately 126MB.

Daily Scan Numbers
Mar 24 327377
Mar 25 436723
Mar 26 406326
Mar 27 410146
Mar 28 371126

Top 10 Scanners

The top 10 scanners were parsed from the logs using the following command to combine the portscan logs into a single file and than organize them according to overall percentage.

Top 10 Source Scanners

%	#	Source
30.95	604075	MY.NET.11.8
20.50	400194	MY.NET.60.43

Top 10 Source Scanners (Cont.)

%	#	Source
7.31	142672	MY.NET.150.113
5.18	101156	MY.NET.150.143
1.45	28364	MY.NET.6.52
1.22	23728	MY.NET.6.50
1.13	22062	MY.NET.6.49
1.12	21954	MY.NET.152.21
1.00	19572	MY.NET.6.45
0.93	18209	64.124.157.32

Top 10 Source Ports

%	**#**	**Port**
30.96	604176	1347
19.28	376241	123
6.86	133804	1257
3.56	69454	7001
3.42	66752	7000
2.66	52007	137
2.26	44142	1057
1.62	31662	0
1.43	27969	6970
1.41	27459	28800

Below is a list of the top 10 Destination IP addresses that were targeted. They are classified by percentage. These IP addresses can give indications of what is being targeted by attackers or what is being used to gather information for a potential future attack.

Top 10 Destinations

%	#	Destination
2.01	39202	MY.NET.1.3
1.70	33152	MY.NET.11.6
1.04	20343	MY.NET.152.20
1.04	20327	MY.NET.152.18
1.04	20301	MY.NET.152.245
1.04	20262	MY.NET.152.10
1.04	20244	MY.NET.152.12
1.04	20215	MY.NET.152.162
1.03	20163	MY.NET.152.14

1.03 20065 MY.NET.152.252

Top 10 Destination Ports

#	Port
604307	1346
180860	4665
125670	80
66764	7001
58976	53
51911	137
47122	7000
30392	6346
26675	0
25714	28800

Top 10 Protocols

%	#	Protocol
88.09	1719160	UDP
11.86	231427	SYN
0.04	723	VECNA
0.01	202	NULL
0.01	125	XMAS
0.00	25	INVALIDACK
0.00	20	UNKNOWN
0.00	8	NOACK
0.00	3	SYNFIN
0.00	2	FULLXMAS
0.00	2	FIN
0.00	1	NMAPID

Top 10 Talkers

Rank	Total # Alerts	Source IP	# Signatures triggered	Destinations involved
rank #1	21640 alerts	MY.NET.153.197	3 signatures	(97 destination IPs)
rank #2	21006 alerts	MY.NET.70.177	2 signatures	(32 destination IPs)
rank #3	16810 alerts	MY.NET.11.6	1 signatures	(48 destination IPs)

rank #4	10805 alerts	66.28.104.154	1 signatures	MY.NET.153.153
rank #5	8911 alerts	MY.NET.11.7	1 signatures	(43 destination IPs)
rank #6	6950 alerts	MY.NET.153.125	3 signatures	(29 destination IPs)
rank #7	6203 alerts	140.142.8.72	1 signatures	MY.NET.153.157
rank #8	6138 alerts	MY.NET.153.203	3 signatures	(13 destination IPs)
rank #9	5087 alerts	MY.NET.150.198	1 signatures	(101 destination IPs)
rank #10	4633 alerts	MY.NET.153.115	2 signatures	(35 destination IPs)
rank #11	3648 alerts	MY.NET.152.19	5 signatures	(24 destination IPs)

#1 Talker

The number one talker in this evaluation, IP address MY.NET.153.197 (aka MY.NET.153.197), is quite possibly infected with a worm of some sort or even physically compromised and being used by an attacker. This is identified by the traffic originating from the system and the fact this system is attempting to connect to IRC. IRC by itself is not "bad" though it is known to be a place where many hackers like to congregate. Many hackers will compromise a system so they can place bots on the system. Especially university systems make great systems to use connections from due to the bandwidth that most major universities have. Bots are scripts that maintain a open connection on IRC and other chat services to allow the hacker to hold their chat channel or other uses such as DOS attacks. The traffic from these attacks or portions there of are attached. The full logs have not been attached due to the enormous size of the logs (over 150 pages just for 1 alert). The ICMP traffic is classified as interesting in this case due to the timing in which it is occurring. It appears the source host is occasionally having problems communicating with the destination host. This raises suspicions and causes interest as the other traffic on the system could be causing this to occur. This would be that the source system has some business continuity issues possibly.

Suspicious or Interesting Traffic from #1 Talker

03/26-10:13:51.431673 [**] INFO Possible IRC Access [**] MY.NET.153.153:2836 -> 195.159.0.91:6667
03/26-10:13:57.470454 [**] INFO Possible IRC Access [**] MY.NET.153.153:2836 -> 195.159.0.91:6667
03/26-10:14:09.520997 [**] INFO Possible IRC Access [**] MY.NET.153.153:2836 -> 195.159.0.91:6667
03/26-10:21:46.182105 [**] INFO Possible IRC Access [**] MY.NET.153.153:2854 -> 65.161.40.3:6667

IRC ACCESS FROM #1 Talker

03/26-08:33:49.665025 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.153.197:1137 -> 207.68.162.250:80
03/26-08:33:49.665025 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.153.197:1137 -> 207.68.162.250:80
03/26-08:36:31.193540 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.153.197:1245 -> 211.32.117.27:80
03/26-08:36:31.193540 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.153.197:1245 -> 211.32.117.27:80

Possible Worm Attacks from #1 Talker

03/26-12:18:35.365945 [**] ICMP Fragment Reassembly Time Exceeded [**] MY.NET.153.153 -> 66.28.104.154
03/26-12:20:04.480535 [**] ICMP Fragment Reassembly Time Exceeded [**] MY.NET.153.153 -> 66.28.104.154
03/26-12:22:34.684874 [**] ICMP Fragment Reassembly Time Exceeded [**] MY.NET.153.153 -> 66.28.104.154
03/26-12:36:35.838817 [**] ICMP Fragment Reassembly Time Exceeded [**] MY.NET.153.153 -> 66.28.104.154

03/26-12:40:21.114298 [**] ICMP Fragment Reassembly Time Exceeded [**] MY.NET.153.153 -> 66.28.104.154
03/26-12:54:36.334274 [**] ICMP Fragment Reassembly Time Exceeded [**] MY.NET.153.153 -> 66.28.104.154

Solution

The solution for this attack would be to apply the vendor patches to the system and clean up any remnants from the attack. The patches and in depth details of the exploits are located at the following URLs. Security Focus gives a detailed explanation of the IIS_UNICODE attack.
http://www.securityfocus.com/bid/1806
Microsoft Patch for this exploit is located at the following location. In addition, some versions of host-based firewalls could assist in preventing these attacks.
http://www.microsoft.com/technet/security/bulletin/MS00-078.asp

#2 Talker

With the current information provided and signatures from this IP address it appears the system is scanning for SNMP servers on the network. This system was detected in the portscans log as well as appears to be triggering large amount of alerts for SNMP. IN addition, there appears to be some traffic that was targeted at this system external to the network. Due to recent exploits, all SNMP traffic should be filtered at the router, unless absolutely needed and even than a ACL should be placed. While some of this traffic could be considered legitimate, the system should be checked to ensure it is now "rooted".

Upon initial examination of the logs, some of the traffic caused concern. There was a TCP SYN scan and than the "attacked" system started Scanning for SNMP servers. While snort may not have identified a signature if the rules were out of date than this should cause some concern. This can be indicative of a compromise though without knowing what was on the system prior the system should be checked and SNMP traffic from external should be potentially blocked.

TCP SCAN TRAFFIC

Mar 24 00:15:02 MY.NET.5.83:28356 -> MY.NET.70.177:7938 SYN ******S*
Mar 24 00:15:02 MY.NET.5.83:28357 -> MY.NET.70.177:7937 SYN ******S*
Mar 24 00:15:27 MY.NET.5.83:14920 -> MY.NET.70.177:7938 SYN ******S*
Mar 24 00:15:27 MY.NET.5.83:14922 -> MY.NET.70.177:7937 SYN ******S*

SNMP SCAN TRAFFIC

03/24-00:20:05.804687 [**] SNMP public access [**] MY.NET.70.177:1068 -> MY.NET.5.31:161
03/24-00:20:05.808693 [**] SNMP public access [**] MY.NET.70.177:1068 -> MY.NET.5.31:161
03/24-00:20:05.814756 [**] SNMP public access [**] MY.NET.70.177:1068 -> MY.NET.5.31:161
03/24-00:20:05.822055 [**] SNMP public access [**] MY.NET.70.177:1068 -> MY.NET.5.31:161

#3 Talker

This talker appears to have possible virus traffic. Due to the unknown factors involved such as Operating system, and applications installed this traffic could be deemed legitimate. However, from the timeframe of accessing the systems and the amount of systems being accessed this traffic appears to be suspicious in nature. The traffic could be attempts to exploit unprotected Windows Network Shares as described in **CERT® Incident Note IN-2000-02**.

03/24-00:01:12.032153 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.213:137
03/24-00:01:35.492258 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.20:137
03/24-00:02:07.304215 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.18:137
03/24-00:02:46.803013 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.21:137
03/24-00:02:47.878803 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.174:137
03/24-00:03:10.795280 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.13:137
03/24-00:03:16.082454 [**] SMB Name Wildcard [**] MY.NET.11.6:137 -> MY.NET.152.46:137

Solution

In order to solve this virus/Trojan problem an up to date anti-virus measure must be implemented. In addition, a network-based Intrusion Detection System can assist in identifying potentially infected systems. To prevent the virus from continually arriving from external sources all traffic that is inbound and directed at ports 137 should blocked, this traffic is not generally needed. If the traffic is necessary, it should only be allowed on a case-by-case basis.

#4 Talker

The fourth talker in this network arrives from an external source. This is suspicious in and of itself since a majority of traffic that the sensor has picked up is from an internal source. The external traffic stands out in this case due to the surrounding activity, as well as the fact that it is from external.

Moreover, this traffic ranks in the top 10 alert category as well as the top 10-source IP addresses.

Portscan Log Traffic

Mar 26 12:15:01 MY.NET.153.153:3779 -> 66.28.104.154:1755 SYN ******S*
Mar 26 12:26:45 MY.NET.153.153:3858 -> 66.28.104.154:1755 UDP
Mar 26 12:26:42 MY.NET.153.153:3949 -> 66.28.104.154:1755 SYN ******S*
Mar 26 12:31:45 MY.NET.153.153:4449 -> 66.28.104.154:1755 SYN ******S*
Mar 26 12:45:27 MY.NET.153.153:4725 -> 66.28.104.154:1755 SYN ******S*
Mar 26 12:46:32 MY.NET.153.153:4768 -> 66.28.104.154:1755 SYN ******S*
Mar 26 12:47:26 MY.NET.153.153:4840 -> 66.28.104.154:1755 SYN ******S*

MISC Large UDP Packet Traffic

03/26-12:15:05.906469 [**] MISC Large UDP Packet [**] 66.28.104.154:1608 -> MY.NET.153.153:3783
03/26-12:15:06.188782 [**] MISC Large UDP Packet [**] 66.28.104.154:1608 -> MY.NET.153.153:3783
03/26-12:15:06.469722 [**] MISC Large UDP Packet [**] 66.28.104.154:1608 -> MY.NET.153.153:3783
03/26-12:15:06.746039 [**] MISC Large UDP Packet [**] 66.28.104.154:1608 -> MY.NET.153.153:3783
03/26-12:15:07.038079 [**] MISC Large UDP Packet [**] 66.28.104.154:1608 -> MY.NET.153.153:3783
03/26-12:15:07.313879 [**] MISC Large UDP Packet [**] 66.28.104.154:1608 -> MY.NET.153.153:3783

From the above traffic, there are indications the traffic was caused by an outside source attempting to access a Windows Media Server on the internal network. This is determined as the traffic that occurs on TCP/1755 and UDP ports 1024-5000. In the files above this traffic is seen.
The information gathered from Microsoft could be the indication of the traffic above.

Server to Client Behind a Firewall (from Microsoft.com)

A firewall configuration that allows users with the Windows Media Player behind a firewall to access Windows Media servers outside the firewall is: Streaming ASF with UDP
Out: TCP on 1755
Out: UDP on 1755
In: UDP between port 1024-5000 (Only open the necessary number of ports.)
Streaming ASF with TCP
In/Out: TCP on port 1755
Streaming ASF with HTTP
In/Out: TCP on Port 80

Solution

A solution to this issue would be to have policies and procedures in place that restrict the streaming of video or music to the internal network. Moreover, the blocking of certain ports in accordance with Microsoft's recommendation and other streaming technology information as can prevent this type of traffic.

#5 Talker

The fifth top talker was MY.NET.11.7. This IP address had generated 8911 alerts. All of the alerts generated by this host were SMB Name Wildcard alerts. Similar to the third largest talker but there was nothing suspicious I could find in regards to this traffic. This may have been a "false positive" alert as there was no additional traffic to indicate that the traffic was malicious. As in the #3 Talker there was additional traffic that could indicate malicious intent, with this host there did not appear to be any malicious traffic. In order to verify this grep was used on the "combinedscans.txt" files to search for occurances of this ip address and any associated traffic in the scans file. Additionally, a scan was run on the alerts file to check and see if the alerts coincided with the scans.

The traffic in those logs appeared to be very normal for the systems on this network.

#5 Talker Traffic that generated Alerts

```
Mar 24 00:34:12 MY.NET.152.185:137 -> MY.NET.11.7:137 UDP
Mar 24 00:34:12 MY.NET.152.185:2855 -> MY.NET.11.7:139 SYN ******S*
Mar 24 00:35:12 MY.NET.152.163:2932 -> MY.NET.11.7:389 UDP
Mar 24 00:35:12 MY.NET.152.163:2930 -> MY.NET.11.7:135 SYN ******S*
Mar 24 00:35:12 MY.NET.152.163:2931 -> MY.NET.11.7:1026 SYN ******S*
Mar 24 00:35:14 MY.NET.152.163:2937 -> MY.NET.11.7:88 UDP
```

Solution

Modification of the the Snort.Conf file, would possibly be in order to better Identify external and internal networks so that alerts are not triggered for normal internal traffic. Since this traffic can be considered normal we could make sure a note is made somewhere for the analyst and let them know this traffic on this host can be ignored.

#6 Talker

The sixth largest talker occurred when the MY.NET.153.125 system attempted multiple connections to destination ports of 515, and port 80 respectively. There were 6950 alerts for this host.

The traffic below is a sample of what triggered the majority of alerts for this host. An initial impression of this traffic was standard printer traffic, and than a "red flag" went up saying virus attack and potential compromise. Upon deeper investigation there is a chance this could be legitimate traffic from a standard host. (Continued below traffic graph)

Traffic that generated Alerts

03/25-11:02:42.334782 [**] connect to 515 from inside [**] MY.NET.153.125:1379 -> MY.NET.150.198:515
03/25-11:02:42.334850 [**] connect to 515 from inside [**] MY.NET.153.125:1379 -> MY.NET.150.198:515
03/25-11:02:42.335385 [**] connect to 515 from inside [**] MY.NET.153.125:1379 -> MY.NET.150.198:515
03/25-11:02:42.335457 [**] connect to 515 from inside [**] MY.NET.153.125:1379 -> MY.NET.150.198:515
03/25-12:09:46.466743 [**] spp_http_decode: CGI Null Byte attack detected [**] MY.NET.153.125:3512 -> 205.188.180.25:80
03/25-12:09:46.466743 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.153.125:3512 -> 205.188.180.25:80
03/25-12:09:46.466743 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.153.125:3512 -> 205.188.180.25:80
03/25-12:10:33.352380 [**] spp_http_decode: CGI Null Byte attack detected [**] MY.NET.153.125:3571 -> 205.188.180.25:80
03/25-12:10:33.352380 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.153.125:3571 -> 205.188.180.25:80
03/25-12:10:33.352380 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.153.125:3571 -> 205.188.180.25:80
03/25-14:32:43.006310 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.153.125:4984 -> 211.233.28.70:80
03/25-14:32:43.006310 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.153.125:4984 -> 211.233.28.70:80
03/25-14:32:43.006310 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.153.125:4984 -> 211.233.28.70:80

(continued)

As was said a few moments ago this traffic could be legitimate when investigating this traffic to determine which virus this was it was found this traffic could be legitimate and a "false positive" traffic created by Simple Chinese characters since Snort does not handle the traffic well (considering it was a different language). Include below is the response that was found to assist in the potential identification of this traffic.

Response found on Messageboard

From: John Berkers

Date: Fri Aug 03 2001 - 04:01:13 CDT
Messages sorted by: [date] [thread] [subject] [author]

The reason you can't find them is that they're actually generated by a preprocessor (http_decode). The http_decode preprocessor normalises any unicode representations of characters and then passes them back to snort for matching against rules. If a particular pattern of unicode characters is detected the ISS Unicode attack event is logged. (no, that's not a spelling error, it doesn't only affect MS IIS, the vuln was first discovered by ISS guys).

You can turn them off by specifying -unicode and -cginull after the http_decode thusly:

preprocessor http_decode: 80 -unicode -cginull

These events are sometimes triggered by visiting sites that use multi-byte characters such as Simplified Chinese etc.

Regards,
John Berkers

When a whois –a <hostname> was accomplished on the IP addresses it was determined there were from APNIC (Asia Pacific NIC). One of the IP addresses were from America Online.

Attached Whois Information

whois -a 205.188.180.25
America Online, Inc (NETBLK-AOL-DTC)
22080 Pacific Blvd
Sterling, VA 20166
US

Netname: AOL-DTC
Netblock: 205.188.0.0 - 205.188.255.255

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

Domain System inverse mapping provided by:

DNS-01.NS.AOL.COM 152.163.159.232
DNS-02.NS.AOL.COM 205.188.157.232

Record last updated on 27-Apr-1998.
Database last updated on 14-Apr-2002 19:58:00 EDT.

---

whois -a 211.233.28.70
Asia Pacific Network Information Center (NETBLK-APNIC-CIDR-BLK)
These addresses have been further assigned to Asia-Pacific users.
Contact info can be found in the APNIC database,
at WHOIS.APNIC.NET or http://www.apnic.net/
Please do not send spam complaints to APNIC.
AU

Netname: APNIC-CIDR-BLK2
Netblock: 210.0.0.0 - 211.255.255.255

Coordinator:
Administrator, System (SA90-ARIN) [No mailbox]
+61 7 3858 3100

Domain System inverse mapping provided by:

NS.APNIC.NET 203.37.255.97
SVC00.APNIC.NET 202.12.28.131
NS.TELSTRA.NET 203.50.0.137
NS.RIPE.NET 193.0.0.193

Regional Internet Registry for the Asia-Pacific Region.
Record last updated on 03-May-2000.
Database last updated on 14-Apr-2002 19:58:00 EDT.

Solution

The solution for this attack would be to disable the preprocessor by commenting out the line in the snort.conf, the previous step is not recommended, or following up and identifying which hosts may have caused this traffic. Moreover, in order to ensure a secure environment a check of the affected system for virii, or other malicious code should be accomplished to ensure that the traffic was not caused by the "malicious code".

#7 Talker

The seventh largest talker in this network comprised of 6203 alerts. This alert is suspicious because this source host is from an external source. In addition, the amount of arriving traffic raises some concerns, as this could be a potential DoS attack. UDP ports tend to be targets for DoS since UDP packets are easily spoofed.

MISC Large UDP Packet

03/25-15:10:14.348581 [**] MISC Large UDP Packet [**] 140.142.8.72:2031 -> MY.NET.153.157:2876
03/25-15:10:14.730738 [**] MISC Large UDP Packet [**] 140.142.8.72:2031 -> MY.NET.153.157:2876
03/25-15:10:15.110267 [**] MISC Large UDP Packet [**] 140.142.8.72:2031 -> MY.NET.153.157:2876
03/25-15:10:16.595781 [**] MISC Large UDP Packet [**] 140.142.8.72:2031 -> MY.NET.153.157:2876
03/25-15:10:16.955523 [**] MISC Large UDP Packet [**] 140.142.8.72:2031 -> MY.NET.153.157:2876
03/25-15:10:18.079422 [**] MISC Large UDP Packet [**] 140.142.8.72:2031 -> MY.NET.153.157:2876
03/25-15:10:18.431200 [**] MISC Large UDP Packet [**] 140.142.8.72:2031 -> MY.NET.153.157:2876
03/25-15:10:18.811774 [**] MISC Large UDP Packet [**] 140.142.8.72:2031 -> MY.NET.153.157:2876
03/25-15:10:20.313061 [**] MISC Large UDP Packet [**] 140.142.8.72:2031 -> MY.NET.153.157:2876
03/25-15:10:20.674878 [**] MISC Large UDP Packet [**] 140.142.8.72:2031 -> MY.NET.153.157:2876
03/25-15:10:21.044088 [**] MISC Large UDP Packet [**] 140.142.8.72:2031 -> MY.NET.153.157:2876

.

This traffic appears on initial glance to be a DoS attack. However, upon closer examination the traffic appears to be legitimate. This was determined by the Destination Port on the system, UDP port 2876. This port is used for a SPS Tunnel. After searching the internet for SPS Tunnels to determine what this service actually was, the SPS Tunnel traffic is a product that is used for VPN tunnels and made by Frontier Technologies. In addition, the source address is registered to another university. While there is still the possibility of this traffic being malicious, if the “destination university” does not have a tunnel of this type than I would approach as malicious traffic and contact the appropriate parties from the source network block.

This alert is generated whenever a UDP packet exceeds 4000 bytes. This rule set is found in the MISC.rules file and the associated rule is:

WHOIS – SOURCE ADDRESS

whois -a 140.142.8.72
NorthWestNet Network Operations Center (NET-UW-SEA)
Academic Computing Center
3737 Brooklyn NE
Seattle, WA 98105
US

Netname: UW-SEA
Netblock: 140.142.0.0 - 140.142.255.255
Maintainer: UWND

Coordinator:
University, Of Washington (OWU2-ARIN) noc@CAC.WASHINGTON.EDU
206-543-5128

Domain System inverse mapping provided by:
HANNA.CAC.WASHINGTON.EDU 140.142.5.5
MARGE.CAC.WASHINGTON.EDU 140.142.5.13
NS.UNET.UMN.EDU 128.101.101.101
Record last updated on 17-Mar-2000.
Database last updated on 14-Apr-2002 19:58:00 EDT.

Solution

The solution for this alert would be to first identify if this is legitimate traffic, without a network diagram, this is not possible in this alert. A check of the rule that caused the alert is necessary to determine if the alert is a true attack or merely communication on the SPS Tunnel (VPN).
To stop the malicious traffic a firewall or router ACL could possibly be necessary to limit this traffic and/or halt it.

#8 Talker

The eighth talker MY.NET.153.203 has some similar traffic to the #6 talker except for the IRC traffic that was identified. There were 3 different signatures that were identified for this host. There were a combined 6138 alerts that this host generated. Upon investigation of the addresses in question there appear to be many accesses to Korean based IP addresses which could be the cause of the spp_http_decode alert that was generated by Snort. Additionally, the connect to port 515 could have been printer traffic as a majority of the traffic is destined for a single host, (possible local printer on same network).
While there was only 2 instances of IRC access this triggers a immediate alert as many systems are compromised for use on IRC services and DoS attacks which are IRC based.

Moreover, the systems were attempting to access Korean IRC servers. There have been many complaints in the message boards recently that Korean Systems have been attempting to access various systems in a unauthorized manner to include virii, malicious code, and others malicious traffic. (As seen on DSHIELD.ORG Messageboards) IRC activity should be blocked based upon policies of the organization to prevent outbound traffic from reaching IRC servers on standard ports.

Signatures for #8 Talker

3 different signatures are present for *MY.NET.153.203* as a source
INFO Possible IRC Access
spp_http_decode: IIS Unicode attack detected
connect to 515 from inside

IRC Signatures from SnortSnarf

03/28-11:30:21.491957 [**] INFO Possible IRC Access [**] MY.NET.153.203:3311 -> 211.63.185.135:6667
03/28-11:32:19.578189 [**] INFO Possible IRC Access [**] MY.NET.153.203:3493 -> 211.192.139.10:6667

IRC Whois Information (single host)

whois -h whois.apnic.net 211.63.185.135
% Rights restricted by copyright. See http://www.apnic.net/db/dbcopyright.html
% (whois6.apnic.net)
inetnum: 211.52.0.0 - 211.63.255.255
netname: KRNIC-KR
descr: KRNIC
descr: Korea Network Information Center
country: KR
admin-c: HM127-AP
tech-c: HM127-AP
remarks: **
remarks: KRNIC is the National Internet Registry
remarks: in Korea under APNIC. If you would like to
remarks: find assignment information in detail
remarks: please refer to the KRNIC Whois DB
remarks: http://whois.nic.or.kr/english/index.html
remarks: **
mnt-by: APNIC-HM
mnt-lower: MNT-KRNIC-AP
changed: hostmaster@apnic.net 20000216
changed: hostmaster@apnic.net 20010606
source: APNIC

person: Host Master
address: 11F, KTF B/D, 1321-11, Seocho2-Dong, Seocho-Gu,
address: Seoul, Korea,137-857
country: KR
phone: +82-2-2186-4500
fax-no: +82-2-2186-4496
e-mail: hostmaster@nic.or.kr
nic-hdl: HM127-AP
mnt-by: MNT-KRNIC-AP
changed: khj@nic.or.kr 20020406
changed: hostmaster@apnic.net 20020415
source: APNIC

inetnum: 211.63.185.0 - 211.63.185.255
netname: KORNET-IDC-JUNGANG-KTIDC-KR
descr: CENTRAL DATA COMMUNICATION OFFICE
descr: 128-9 YEUNKEONDONG JONGROKU
descr: SEOUL
descr: 110-460
country: KR
admin-c: GP960-KR
tech-c: WK2986-KR
remarks: This IP address space has been allocated to KRNIC.
remarks: For more information, using KRNIC Whois Database
remarks: whois -h whois.nic.or.kr
remarks: This information has been partially mirrored by APNIC from
remarks: KRNIC. To obtain more specific information, please use the
remarks: KRNIC whois server at whois.krnic.net.
mnt-by: MNT-KRNIC-AP
changed: hostmaster@nic.or.kr 20020408
source: KRNIC

person: GilSoon Park
country: KR
phone: +82-2-747-9213
fax-no: +82-2-766-5901
e-mail: gspark@kornet.net
nic-hdl: GP960-KR
remarks: This information has been partially mirrored by APNIC from
remarks: KRNIC. To obtain more specific information, please use the
remarks: KRNIC whois server at whois.krnic.net.
mnt-by: MNT-KRNIC-AP
changed: hostmaster@nic.or.kr 20020408
source: KRNIC

Solution:
In order to resolve this type of activity filters or firewalls can assist in preventing routing of traffic destined for port 6667-6669, which many IRC servers run on. An audit of systems to prevent IRC clients from being installed can be implemented, using one of the many host-based IDS products. Since this is a university, this may not be feasible though since many universities support the IRC service.

#9 Talker

The ninth most frequent talker on the network was MY.NET.150.198. This IP address generated 5087 alerts directed at 101 different destination IP addresses. There was only a single signature detected for this IP address. The signature that was picked up belonged to SNMP public Access. This could be a recon attack or possible normal traffic depending on the system usage.

Alerts Generated from #9 Talker (sampled)

03/24-00:00:03.006899 [**] SNMP public access [**] MY.NET.150.198:1025 -> MY.NET.113.202:161
03/24-00:05:36.823635 [**] SNMP public access [**] MY.NET.150.198:1025 -> MY.NET.151.114:161
03/24-00:05:36.823750 [**] SNMP public access [**] MY.NET.150.198:1025 -> MY.NET.151.114:161
03/24-00:08:03.024891 [**] SNMP public access [**] MY.NET.150.198:1025 -> MY.NET.113.202:161
03/24-00:12:02.996153 [**] SNMP public access [**] MY.NET.150.198:1025 -> MY.NET.113.202:161
03/24-00:15:36.826920 [**] SNMP public access [**] MY.NET.150.198:1025 -> MY.NET.151.114:161
03/24-00:15:36.827035 [**] SNMP public access [**] MY.NET.150.198:1025 -> MY.NET.151.114:161
03/24-00:20:03.016907 [**] SNMP public access [**] MY.NET.150.198:1025 -> MY.NET.113.202:161
03/24-00:24:03.022128 [**] SNMP public access [**] MY.NET.150.198:1025 -> MY.NET.113.202:161
03/24-00:25:36.867279 [**] SNMP public access [**] MY.NET.150.198:1025 -> MY.NET.151.114:161

03/24-00:25:36.867392 [**] SNMP public access [**] MY.NET.150.198:1025 -> MY.NET.151.114:161
03/24-00:28:03.025674 [**] SNMP public access [**] MY.NET.150.198:1025 -> MY.NET.113.202:161
03/24-00:32:03.029161 [**] SNMP public access [**] MY.NET.150.198:1025 -> MY.NET.113.202:161

The traffic that was generated could be a compromised system, scanning for additional systems to compromise. This information is based upon the recently announced security vulnerabilities with SNMP. Additionally it appears the source system is scanning the network seeking systems with public access, which many times is misconfigured and allows write access or even read access to gather usernames or other vital information. Even though there is no "indication" of the inbound traffic, there is evidence of outbound traffic from this system. This system should be checked to ensure it is not compromised. This rule appears to have been written by the administrator of the IDS, as while performing a search on the rules and conf files for Snort there were no standard rules for this traffic.
This traffic does not appear to have targeted any systems external to the network. With the current provided alerts it is unknown if there were successful connections to other systems.

Solution

In order to solve this traffic all public community passwords should be set and the community strings should be changed. In addition, all external SNMP traffic should be blocked at the perimeters to the network, unless necessary. This prevents unauthorized overflows or reconnaissance attacks from external to your network.

(the #10 top talker will be an evaluation of the #11 Talker traffic due to #10 traffic is the same or very closely related to previous talker traffic. The only reason it ended up in the #10 spot was due to the sheer volume while #11 was shortly behind)

#10 Talker-but actually analysis of #11

There were 5 different signatures present for MY.NET.152.19 as a source which consisted of 4 instances of INFO Possible IRC Access, 72 instances of ICMP Echo Request Nmap or HPING2, 522 instances of ICMP Echo Request L3retriever Ping, 525 instances of SMB Name Wildcard, 2525 instances of spp_http_decode: IIS Unicode attack detected. Some additional suspicious traffic consisted of
4 different signatures are present for 192.168.152.19 as a destination, 4 instances of High port 65535 udp - possible Red Worm – traffic, 6 instances of INFO - Possible Squid Scan, 6 instances of SCAN Proxy attempt, 525 instances of SMB Name Wildcard.

Traffic departing from MY.NET.152.19 (sampled)

03/27-13:05:04.805161 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.152.19:3166 -> 202.30.143.18:80
03/27-13:05:04.805161 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.152.19:3166 -> 202.30.143.18:80
03/27-13:05:04.805161 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.152.19:3166 -> 202.30.143.18:80
03/27-13:10:15.505119 [**] ICMP Echo Request L3retriever Ping [**] MY.NET.152.19 -> MY.NET.11.7
03/27-13:10:15.505533 [**] SMB Name Wildcard [**] MY.NET.152.19:137 -> MY.NET.11.7:137
03/27-13:10:17.788737 [**] ICMP Echo Request L3retriever Ping [**] MY.NET.152.19 -> MY.NET.11.5
03/27-13:10:17.789945 [**] SMB Name Wildcard [**] MY.NET.152.19:137 -> MY.NET.11.5:137
03/27-13:12:13.753585 [**] INFO Possible IRC Access [**] MY.NET.152.19:3405 -> 211.216.53.129:6667
03/27-13:12:17.571737 [**] INFO Possible IRC Access [**] MY.NET.152.19:3405 -> 211.216.53.129:6667
03/27-13:12:23.917210 [**] INFO Possible IRC Access [**] MY.NET.152.19:3405 -> 211.216.53.129:6667
03/27-13:14:28.731460 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.152.19:3410 -> 211.233.53.251:80
03/27-13:14:28.731460 [**] spp_http_decode: IIS Unicode attack detected [**] MY.NET.152.19:3410 -> 211.233.53.251:80

Traffic arriving at MY.NET.152.19 (sampled)

03/27-13:44:01.836764 [**] SMB Name Wildcard [**] MY.NET.11.7:137 -> MY.NET.152.19:137
03/27-13:44:06.977147 [**] SMB Name Wildcard [**] MY.NET.11.5:137 -> MY.NET.152.19:137
03/27-13:49:07.749614 [**] SMB Name Wildcard [**] MY.NET.11.7:137 -> MY.NET.152.19:137
03/27-13:49:10.597221 [**] SMB Name Wildcard [**] MY.NET.11.5:137 -> MY.NET.152.19:137
03/27-14:00:53.869108 [**] SMB Name Wildcard [**] MY.NET.11.7:137 -> MY.NET.152.19:137
03/27-14:05:46.638548 [**] SMB Name Wildcard [**] MY.NET.11.7:137 -> MY.NET.152.19:137
03/27-14:08:47.064285 [**] High port 65535 udp - possible Red Worm - traffic [**] MY.NET.6.60:48508 -> MY.NET.152.19:65535

03/27-14:08:51.909312 [**] SMB Name Wildcard [**] MY.NET.11.7:137 -> MY.NET.152.19:137
03/27-14:09:09.395392 [**] SCAN Proxy attempt [**] 195.22.174.130:42854 -> MY.NET.152.19:8080
03/27-14:09:09.545532 [**] INFO - Possible Squid Scan [**] 195.22.174.130:42857 -> MY.NET.152.19:3128
03/27-14:09:25.766368 [**] SCAN Proxy attempt [**] 213.226.142.114:4681 -> MY.NET.152.19:8080
03/27-14:09:25.767341 [**] INFO - Possible Squid Scan [**] 213.226.142.114:4679 -> MY.NET.152.19:3128
03/27-14:09:26.694833 [**] INFO - Possible Squid Scan [**] 213.226.142.114:4679 -> MY.NET.152.19:3128
03/27-14:09:27.663691 [**] SCAN Proxy attempt [**] 213.226.142.114:4681 -> MY.NET.152.19:8080
03/27-14:09:27.746700 [**] INFO - Possible Squid Scan [**] 213.226.142.114:4679 -> MY.NET.152.19:3128
03/27-14:10:24.196610 [**] INFO - Possible Squid Scan [**] 217.39.139.35:33595 -> MY.NET.152.19:3128
03/27-14:10:24.201374 [**] SCAN Proxy attempt [**] 217.39.139.35:33457 -> MY.NET.152.19:8080
03/27-14:10:24.721962 [**] INFO - Possible Squid Scan [**] 217.39.139.35:33595 -> MY.NET.152.19:3128
03/27-14:10:24.728012 [**] SCAN Proxy attempt [**] 217.39.139.35:33457 -> MY.NET.152.19:8080
03/27-14:10:25.353849 [**] SCAN Proxy attempt [**] 217.39.139.35:33827 -> MY.NET.152.19:8080

The first alert that was investigated was the spp_http_decode alert that occurs 3/27 at approximately 13:05. In order to check if this is potentially malicious traffic we would check a quick whois, and see if the alert could be caused by the simple Chinese characters as mentioned earlier by John Berker in his email. This was the case with this alert it appears to be due to the user surfing the web to a system which displays these characters on their website and triggers this alert.
The second alert that is very concerning would be the L3retriever Ping which occurred from the system and was directed at MY.NET.11.7. This traffic means the system could actively be seeking a exploit on the network. This automatically raises the red flags. While there is no previous evidence this specific system is compromised, this traffic could be accomplished by an "insider" or someone who has legitimate access to the system. The possibility is still that the intruder managed to bypass the IDS system. Moreover, some of the additional traffic points out this system could be being used as a staging ground for attacks. In conjunction with this located within the traffic that arrived to this system there appears to be an attempt to access UDP 65535. This is a very high port number in fact is the end of the line and many Trojans attempt to access and connect on higher ports where port scanners do not "normally" check for services, as well as Administrators sometimes fail to check. All these conditions should be checked to ensure that the system is not compromised and not dishing out attacks.

Rule for detecting L3Retriever Ping

```
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"ICMP L3retriever
Ping"; content: "ABCDEFGHIJKLMNOPQRSTUVWABCDEFGHI"; itype: 8; icode: 0;
depth: 32; reference:arachnids,311; classtype:attempted-recon; sid:466;
rev:1;)
```

Nmap Ping/HPING

03/24-00:42:23.745499 [**] ICMP Echo Request Nmap or HPING2 [**] MY.NET.152.19 -> MY.NET.11.7
03/24-02:35:26.661578 [**] ICMP Echo Request Nmap or HPING2 [**] MY.NET.152.19 -> MY.NET.11.7
03/24-04:14:31.657176 [**] ICMP Echo Request Nmap or HPING2 [**] MY.NET.152.19 -> MY.NET.11.7
03/24-05:48:34.535865 [**] ICMP Echo Request Nmap or HPING2 [**] MY.NET.152.19 -> MY.NET.11.7
03/24-07:26:37.454666 [**] ICMP Echo Request Nmap or HPING2 [**] MY.NET.152.19 -> MY.NET.11.7
03/24-09:22:40.372702 [**] ICMP Echo Request Nmap or HPING2 [**] MY.NET.152.19 -> MY.NET.11.7
03/24-11:00:43.179532 [**] ICMP Echo Request Nmap or HPING2 [**] MY.NET.152.19 -> MY.NET.11.7
03/24-12:37:45.976233 [**] ICMP Echo Request Nmap or HPING2 [**] MY.NET.152.19 -> MY.NET.11.7
03/24-14:16:48.885412 [**] ICMP Echo Request Nmap or HPING2 [**] MY.NET.152.19 -> MY.NET.11.7
03/24-16:10:51.833988 [**] ICMP Echo Request Nmap or HPING2 [**] MY.NET.152.19 -> MY.NET.11.7
03/24-17:44:54.740409 [**] ICMP Echo Request Nmap or HPING2 [**] MY.NET.152.19 -> MY.NET.11.7
03/24-19:31:57.654073 [**] ICMP Echo Request Nmap or HPING2 [**] MY.NET.152.19 -> MY.NET.11.7

Solution

In order to efficiently prevent damage to any system all unnecessary services should be shutdown, as well as patches on the active services up to date. Moreover, host-based IDS can be placed on systems to let administrators know when modifications have been made to critical files. An up-to-date IDS should be in place on the network to let the administrator know when malicious traffic is occurring.

5 External Addresses

# Of Attacks	Source	Destination	Method
40	24.206.27.148	192.168.5.96	WWEB-IIS view source via translate header
14	130.243.48.100	192.168.150.6	WEB-MISC Attempt to execute cmd
12	217.120.35.172	192.168.153.159	Null scan!
9	212.179.127.56	192.168.150.133	Watchlist 000220 IL-ISDNNET-990517
7	172.150.50.154	192.168.5.96	WEB-CGI scriptalias access

For the external addresses, Microsoft Excel was used to sort the data and than remove all the MY.NET hosts. After this was completed the data was sorted again by number of

attacks and by Method. Than the top 5 attackers from each different attack method was used.

Registration Information

IP : 24.206.27.148

root [/]: whois -a 24.206.27.148
GS Communications (NETBLK-GSCOMM-1BLK)
442 West Patrick Street
Frederick, MD 21701
US
Netname: GSCOMM-1BLK
Netblock: 24.206.0.0 - 24.206.31.255
Maintainer: GSCA
Coordinator:
Sanders, Matthew (MS179-ARIN) msanders@gscommunications.com
301-662-6822

Domain System inverse mapping provided by:
DNS1.GSCYCLONE.COM 209.36.53.10
DNS2.GSCYCLONE.COM 209.36.53.55
ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE

Record last updated on 24-Jan-2002.
Database last updated on 16-Apr-2002 19:59:02 EDT.

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

root [/]: nslookup 24.206.27.148
Server: union.MYDNS.net
Address: 216.XXX.YYY.150

*** union.MYDNS.net can't find 24.206.27.148: Non-existent host/domain
root [/]:

IP Address: 130.243.48.100

root [/]: whois -a 130.243.48.100
European Regional Internet Registry/RIPE NCC (NET-SUNETREGAB-RIPE)
These addresses have been further assigned
to European users. Contact information can
be found in the RIPE database at whois.ripe.net
NL

Netname: SUNETREGAB-RIPE
Netblock: 130.242.0.0 - 130.243.255.255
Maintainer: RIPE
Coordinator:
Reseaux IP European Network Co-ordination Centre Singel 258 (RIPE-NCC-ARIN)
nicdb@RIPE.NET
+31 20 535 4444

Domain System inverse mapping provided by:
SUNIC.SUNET.SE 192.36.125.2
FALUN.DNS.SWIP.NET 192.71.220.13
NS.RIPE.NET 193.0.0.193

Record last updated on 29-Mar-2000.
Database last updated on 16-Apr-2002 19:59:02 EDT.
The ARIN Registration Services Host contains ONLY Internet
Network Information: Networks, ASN's, and related POC's.
Please use the whois server at rs.internic.net for DOMAIN related
Information and whois.nic.mil for NIPRNET Information.

root [/]: whois -h whois.ripe.net 130.243.48.100
% This is the RIPE Whois server.
% The objects are in RPSL format.
% Please visit http://www.ripe.net/rpsl for more information.
% Rights restricted by copyright.
% See http://www.ripe.net/ripencc/pub-services/db/copyright.html

inetnum: 130.243.32.0 - 130.243.63.255
netname: SE-DU
descr: Dalarna University
country: SE
admin-c: ANNO1-RIPE
tech-c: ANNO1-RIPE
status: ASSIGNED PA
remarks: for abuse-matters contact abuse@du.se
mnt-by: SUNET-MNT
changed: fredrik@sunet.se 19981202
changed: fredrik@sunet.se 20000711
source: RIPE
route: 130.243.32.0/19
descr: Dalarna University
origin: AS2834
mnt-by: SUNET-MNT
changed: fredrik@sunet.se 19981207
source: RIPE

person: Anders Nordahl
address: Dalarna University
address: S-781 88 Borlange, SWEDEN
phone: +46 23 778122
fax-no: +46 23 778050
e-mail: ano@du.se
nic-hdl: ANNO1-RIPE
changed: fredrik@sunet.se 19981207
source: RIPE

IP Address: 217.120.35.172

root [/]: whois -a 217.120.35.172
European Regional Internet Registry/RIPE NCC (NET-217-RIPE)
These addresses have been further assigned
to European users. Contact information can
be found in the RIPE database at whois.ripe.net
NL

Netname: 217-RIPE
Netblock: 217.0.0.0 - 217.255.255.255
Maintainer: RIPE

Coordinator:
Reseaux IP European Network Co-ordination Centre Singel 258 (RIPE-NCC-ARIN)
nicdb@RIPE.NET
+31 20 535 4444

Domain System inverse mapping provided by:

NS.RIPE.NET 193.0.0.193
NS.EU.NET 192.16.202.11
AUTH00.NS.UU.NET 198.6.1.65
NS3.NIC.FR 192.134.0.49
SUNIC.SUNET.SE 192.36.125.2
MUNNARI.OZ.AU 128.250.1.21
NS.APNIC.NET 203.37.255.97
SVC00.APNIC.NET 202.12.28.131

Record last updated on 05-Jun-2000.
Database last updated on 16-Apr-2002 19:59:02 EDT.

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

root [/]: whois -h whois.ripe.net 217.120.35.172
% This is the RIPE Whois server.
% The objects are in RPSL format.
% Please visit http://www.ripe.net/rpsl for more information.
% Rights restricted by copyright.
% See http://www.ripe.net/ripencc/pub-services/db/copyright.html

inetnum: 217.120.32.0 - 217.120.47.255
netname: BENELUX-PALET-DBSCH-3
descr: @Home Benelux Headend block
country: NL
admin-c: ABNO1-RIPE
tech-c: ABIM3-RIPE
remarks: For abuse issues, please email abuse@corp.nl.home.com
status: ASSIGNED PA
mnt-by: BENELUX-MNT
mnt-lower: BENELUX-MNT
changed: judithh@excitehome.net 20010605
source: RIPE

route: 217.120.0.0/14
descr: @Home Benelux
origin: AS9143
mnt-by: BENELUX-MNT
changed: judithh@corp.home.net 20010103
source: RIPE

role: AtHome Benelux Network Operations Centre
address: Gyroscoopweg 90-92
address: 1042 AX Amsterdam
address: The Netherlands
phone: +31 20 885 5544
fax-no: +31 20 885 5525
e-mail: noc@corp.nl.home.com
trouble: reports of network abuse, pls. contact
trouble: abuse@corp.nl.home.com
admin-c: JVV19-RIPE
tech-c: JH4485-RIPE
tech-c: RCE3-RIPE
nic-hdl: ABNO1-RIPE
notify: ipmgmt@corp.nl.home.com
changed: judithh@excitehome.net 20010503
source: RIPE

role: AtHome Benelux IP Mgmt
address: Gyroscoopweg 90-92

address: 1042 AX Amsterdam
address: The Netherlands
phone: +31 20 885 5544
fax-no: +31 20 885 5525
e-mail: ipmgmt@excitehome.net
trouble: reports of network abuse, pls. contact
trouble: abuse@corp.nl.home.com
admin-c: JH4485-RIPE
tech-c: JH4485-RIPE
tech-c: RCE3-RIPE
nic-hdl: ABIM3-RIPE
notify: judithh@excitehome.net
changed: judithh@excitehome.net 20010503
source: RIPE

IP Address: 212.179.127.56

root [/]: whois -a 212.179.127.56
European Regional Internet Registry/RIPE NCC (NET-RIPE-NCC-)
These addresses have been further assigned to European users.
Contact info can be found in the RIPE database, via the
WHOIS and TELNET servers at whois.ripe.net, and at
http://www.ripe.net/perl/whois/
NL

Netname: RIPE-NCC-212
Netblock: 212.0.0.0 - 212.255.255.255
Maintainer: RIPE

Coordinator:
Reseaux IP European Network Co-ordination Centre Singel 258 (RIPE-NCC-ARIN)
nicdb@RIPE.NET
+31 20 535 4444
Domain System inverse mapping provided by:

NS.RIPE.NET 193.0.0.193
NS.EU.NET 192.16.202.11
AUTH03.NS.UU.NET 198.6.1.83
NS2.NIC.FR 192.93.0.4
SUNIC.SUNET.SE 192.36.125.2
MUNNARI.OZ.AU 128.250.1.21
NS.APNIC.NET 203.37.255.97

To search on arbitrary strings, see the Database page on
the RIPE NCC website at http://www.ripe.net/perl/whois/

Record last updated on 16-Oct-1998.

Database last updated on 16-Apr-2002 19:59:02 EDT.

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

root [/]: whois -h whois.ripe.net 212.179.127.56
% This is the RIPE Whois server.
% The objects are in RPSL format.
% Please visit http://www.ripe.net/rpsl for more information.
% Rights restricted by copyright.
% See http://www.ripe.net/ripencc/pub-services/db/copyright.html

inetnum: 212.179.127.0 - 212.179.127.127
netname: ARAVA-DEVELOPMENT-COMPANY-LTD
descr: ARAVA-DEVELOPMENT-LAN
country: IL
admin-c: ES4966-RIPE
tech-c: NP469-RIPE
status: ASSIGNED PA
notify: hostmaster@isdn.net.il
mnt-by: RIPE-NCC-NONE-MNT
changed: hostmaster@isdn.net.il 20000525
source: RIPE

route: 212.179.0.0/17
descr: ISDN Net Ltd.
origin: AS8551
notify: hostmaster@isdn.net.il
mnt-by: AS8551-MNT
changed: hostmaster@isdn.net.il 19990610
source: RIPE

person: Eran Shchori
address: BEZEQ INTERNATIONAL
address: 40 Hashacham Street
address: Petach-Tikva 49170 Israel
phone: +972 3 9257710
fax-no: +972 3 9257726
e-mail: hostmaster@bezeqint.net
nic-hdl: ES4966-RIPE
changed: registrar@ns.il 20000309
source: RIPE

person: Nati Pinko

address: Bezeq International
address: 40 Hashacham St.
address: Petach Tikvah Israel
phone: +972 3 9257761
e-mail: hostmaster@isdn.net.il
nic-hdl: NP469-RIPE
changed: registrar@ns.il 19990902
source: RIPE

IP Address: 172.150.50.154

root [/]: whois -a 172.150.50.154
America Online, Inc. (NETBLK-AOL-172BLK)
12100 Sunrise Valley Drive
Reston, VA 20191
US

Netname: AOL-172BLK
Netblock: 172.128.0.0 - 172.191.255.255
Maintainer: AOL

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

Domain System inverse mapping provided by:

DAHA-01.NS.AOL.COM 152.163.159.233
DAHA-02.NS.AOL.COM 205.188.157.233

ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE

Record last updated on 28-Mar-2001.
Database last updated on 16-Apr-2002 19:59:02 EDT.

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

Last Updated: October 18th, 2017

Upcoming Training

SANS Berlin 2017	Berlin, Germany	Oct 23, 2017 - Oct 28, 2017	Live Event
San Diego Fall 2017 - SEC503: Intrusion Detection In-Depth	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	vLive
SANS San Diego 2017	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	Live Event
SANS Seattle 2017	Seattle, WA	Oct 30, 2017 - Nov 04, 2017	Live Event
SANS Paris November 2017	Paris, France	Nov 13, 2017 - Nov 18, 2017	Live Event
Community SANS Pensacola SEC503	Pensacola, FL	Nov 27, 2017 - Dec 02, 2017	Community SANS
SIEM & Tactical Analytics Summit & Training	Scottsdale, AZ	Nov 28, 2017 - Dec 05, 2017	Live Event
SANS Munich December 2017	Munich, Germany	Dec 04, 2017 - Dec 09, 2017	Live Event
SANS Cyber Defense Initiative 2017	Washington, DC	Dec 12, 2017 - Dec 19, 2017	Live Event
SANS Security East 2018	New Orleans, LA	Jan 08, 2018 - Jan 13, 2018	Live Event
Community SANS Nashville SEC401^	Nashville, TN	Jan 08, 2018 - Jan 13, 2018	Community SANS
Las Vegas 2018 - SEC503: Intrusion Detection In-Depth	Las Vegas, NV	Jan 28, 2018 - Feb 02, 2018	vLive
SANS Las Vegas 2018	Las Vegas, NV	Jan 28, 2018 - Feb 02, 2018	Live Event
SANS London February 2018	London, United Kingdom	Feb 05, 2018 - Feb 10, 2018	Live Event
SANS Dallas 2018	Dallas, TX	Feb 19, 2018 - Feb 24, 2018	Live Event
SANS Northern VA Spring - Tysons 2018	McLean, VA	Mar 17, 2018 - Mar 24, 2018	Live Event
SANS Secure Canberra 2018	Canberra, Australia	Mar 19, 2018 - Mar 24, 2018	Live Event
SANS 2018	Orlando, FL	Apr 03, 2018 - Apr 10, 2018	Live Event
SANS Baltimore Spring 2018	Baltimore, MD	Apr 21, 2018 - Apr 28, 2018	Live Event
SANS Security West 2018	San Diego, CA	May 11, 2018 - May 18, 2018	Live Event
Community SANS Columbia SEC503	Columbia, MD	Aug 13, 2018 - Aug 18, 2018	Community SANS
SANS OnDemand	Online	Anytime	Self Paced
SANS SelfStudy	Books & MP3s Only	Anytime	Self Paced