Global Information Assurance Certification Paper

SANS Intrusion Detection in Depth

GCIA Practical V.3.2

Prepared by David Zamler

Table of Contents

Assignment 1: Describe the State of Intrusion Detection

Introduction

Intrusion Detection Systems (IDS) has come to prominence the last few years as information security related attacks flourished with the proliferation of distributed engineering and computing. IDS adds a different dimension to defense-in-depth by detecting both external attacks and internal misuse of computing resources. IDS through automated processes, often provides active, real time defense against attacks or misuse. This feature makes IDS attractive to complement other existing information security products such as firewall.

When selecting IDS, because it is fairly new, evaluation criteria and standards are not as common and mature as other information security technologies. This paper will attempt to identify and discuss a set of criteria that ought to be included when evaluating a network based IDS. (Host or application based IDS will not be in the scope of this paper.) As well, a description and objective of these criteria will also be provided which may be used to form the basis of a selection matrix.

Factors Affecting The Selection Of IDS

Total Cost of Ownership

One term that has enjoyed increased popularity is the Total Cost of Ownership (TCO). This paper will not discuss TCO in any depth. It is simply because different organizations have different measurements for TCO. One simple model for TCO is to categorize it into three main components: Acquisition, Deployment and Management. Acquisition mainly deals with the cost structure of the IDS. For example, is the IDS competitively priced when compared to its peers in the industry? Is IDS available as a managed service? These are some of the factors that required careful consideration. Deployment entails how the IDS are to be rolled out and its compatibility to the current organization's computing environment. Some of the examples include: is the monitoring and reporting component a desktop native operating system or ported application, requires non-standard operating system or runs in web browser? What is the total time from unpack to operational deployment? etc. Management component within TCO is how the IDS will be managed and what built-in tools are available to handle such tasks. Factors such as whether the IDS sensor allows remote diagnostic tools and automated updates of operating system, driver, management and application software can be handled by the sensor should be considered.

Human Resource

When forming a selection matrix, there are certain requirements that cannot be addressed. These requirements include the human resources available and management's operational

requirements. For instance if the turnover rate of new employees is high in the network and security organization, training new staff to help support the IDS product may be a contributing factor in the selection of the IDS. If a particular IDS requires specialized skills to maintain, training costs will increase when compared to a product requiring fewer specialized skills. Also the reliance on a specific individual may occur in this situation because the knowledge of the IDS product may not be easily transferable. Additionally, if the time that an employee has to maintain each security product within the organization is limited, then a product with a significant cost of maintenance may not be a good choice. These factors should be weighed carefully.

Vendor Support and Maintenance

Vendor commitment to product updates and product support is another business concern that may not be well reflected in the criteria in this paper. The history of the IDS industry is relatively short. Therefore, there is little data regarding the support and maintenance performance of the vendors in the industry. Also, if the need for an IDS product is immediate and implementation must occur in a relatively short period of time, set-up and configuration time should be an identified component.

Selection Criteria

How are the selection criteria formed? The selection criteria are constructed based on eleven critical areas that range from compliance to open IDS standards to organizational specific standards and requirements on IDS architecture, analysis capabilities, management, monitoring, and support. The following are described below in detail:

1. Compliance to Existing IDS Standards

For selection of any technology, compliance to existing product group standards is a necessity. The objective of these criteria is to ensure the IDS selected for evaluation-achieved standardization across current open source forums and receptive to incorporate these standards. Some of the most important open source forums include:

- Internet Engineering Task Force (IETF)
- Intrusion Detection Exchange Working Group (IDWG)
- Common Intrusion Detection Framework (CIDF)
- Checkpoint's Open platform for Security (OPSEC)
- National Institute of Standards and Technology (NIST)

Links to the aforementioned organizations are listed in the Reference section below.

2. Support Current Technology Standards

The purpose of these criteria is to ensure the IDS selected will conform to the organization's current technology standards. Interoperability with other platforms is critical so that the IDS evaluated may comply with the organization's general technology relationships and directions. No examples will be given, as different organizations will differ in platforms used.

3. Architecture Requirement

The objective of this is to outline what the IDS should support at a minimum. From an architecture point of view, it should have the flexibility to be implemented in an enterprise environment. As well, it should contain strong management and authentication services. This includes provisions that enforce 'segregation of duties' between IDS administrators and authentication controls for users and components of the entire system. Further all requirements should be vendor agnostic. IDS architecture should support:

- Secure method of communication between IDS and collector (i.e. SSL/3DES/encrypted tunnel)
- Contain a distributed management/monitoring capability. Specifically, the sensor, management console and GUI should all be separate systems
- Supports central administration functions that can control multiple sensors
- Ability to parse administration functions among multiple IDS administrators with access controls (i.e. can limit IDS administrators access by management console system or sensor system)
- Contains appropriate authentication mechanisms (i.e. all IDS administrators have their own user id and password).
- Authentication between the sensor and console should occur prior to communication between the two components
- Analysis of packet information occurs at the sensor
- Analysis of packet information can be configured to occur at an analysis engine separate from the packet capture system
- Central Console and GUI may be on separate systems
- Multiple, simultaneous GUI's
- A logging mechanism that can be separated from the data analysis and data capture engine either physically or logically
- Interaction with Firewall and Router

4. Packet Capture Capability

The objective of these criteria is to evaluate basic functions and commonalities shared by IDS technologies. This for example would include processing speeds, supported network interfaces, and sensor critical mass/failure notifications. The capability that the IDS evaluated should comprised of the following:

- Built-in Defrag Processor
- Statistical analysis for abnormal activity patterns
- Operating-system audit trail management with recognition of user activity
- Packet Capture occurs in Real-time

- Packet Capture Supports 10/100/1000 bit speeds
- Packet Capture supports Multiple Nics Min. 2 per sensor
- Notification from the IDS when packets are dropped and at what capacity level they are dropped

5. Packet Analysis Capability

The objective of this criterion is to ensure the IDS possess the ability to benchmark and analyze traffic patterns and behavior. In addition, this criterion also attempts to determine the robustness and complexity of IDS software and software architecture. The capability that the IDS evaluated should have the following characteristics:

- Batch or Interval Oriented
- Real-time Analysis of packet is offered
- Statistical Analysis - Finds deviations from normal patterns of behavior against RFC STD
- Acceptance of 3rd party signatures
- Ability to customize signatures
- Uses string matching for signature detection
- String matching signatures are extensible
- Built and based upon a multi-threaded application
- Programming language for the analysis engine is common and not proprietary
- Statistical Analysis - Analysis extends beyond IP Header and including TCP/UDP and ICMP
- Reassembling of IP packets are done by the analysis engine
- Reassembling of TCP packets and all protocols within TCP are done by the analysis engine
- Reassembling of UDP packets and all protocols within UDP are done by the analysis engine
- Support of SMP
- Ability for the IDS to identify its critical mass (i.e. 100,000 concurrent sessions) of reliability and effectiveness.

6. Signature Set Content and Presentation

Signature set should present pertinent information in order to allow for analysis of the attack signature. The following is a list of information that the signature set should offer:

- Signature Analysis - Pattern matching against database of known attacks from open source

signature community

- Signatures are ranked as high, medium and low risk. Definition should be provided for each classification
- Signatures should classify to identify different types of activities. For example, attacks, suspicious activity, protocol anomaly and network events
- Signatures are grouped by the type system effected by the exploit or vulnerability
- Signatures to include a description, explanation of the trigger and possible resolution of the event
- Signatures to include BUGTRAQ and CVE numbers

7. Data Forensic/Logging Capability

Data-forensics and logging capability criteria is to evaluate the IDS logging options and how flexible the product is with integrating with other third party log aggregation/correlation tools. It also tests the IDS's ability to support verbose and specific playback functions post-ex-facto an event. The capability that the IDS evaluated should include the following:

- Supports the ability to log events in a secured fashion such as writing a log file to a printer or CD-ROM
- Interoperable with and supported by third party log aggregation/correlation tool to receive logs and alerts
- Supports the ability to store 'interesting' network information without utilizing too much hard drive space by providing the capability of filtering logged data
- Supports a session capture capability that can record specific events such as any transaction occurring over the network that originates from a specific IP address or user
- Possess the functions of visual session playback and full packet capture
- Ability to have recorded events be replayed to show keystrokes

8. Reporting Capability

Reporting criteria measures the IDS with respect to reporting delivery, aggregation, statistical and trend analysis, default templates, tamperproof controls, and recommendation on reporting alerts. The format of the report should be easy to read and understand by the user. The capability that the IDS evaluated should consist of the following:

- Reports are easy to read and understand
- Reports support graphical representations of data
- Supports the ability to schedule the automatic generation of reports
- Supports the scheduling of a report to be generated and sent to a printer or e-mailed to an individual

- Contains sufficient explanations or definitions of the attack signatures in 'normal' language
- Contains fixes for attack signatures
- Ability to allow the administrator to modify reports by date, IP address or network segment
- Ability to allow the administrator to create new reports
- Ability to consolidate data from other reports
- Ability to consolidate report data over time regardless of current software release of the IDS product
- Ability to export data to database or external source (i.e.. ANSI, CSV, Crystal Reports)
- Ability to report to different layers of detail for organizational roles
- Log files have built-in tamper-proof security controls to detect modification
- Ability to report and audit on IDS Policy and IDS changes from sensor and console
- Timeliness of intrusion's occurrence and its timely reported

9. Alert Capability

The objective of the alert capability is to ensure that IDS technology supports a wide array of alerting options and mechanisms over multiple communication channels. This will allow alerts be integrated with a variety of existing workflow measures on incident response. The capability that the IDS evaluated should include the following:

- Alerts should be sent from the central console
- Supports the ability to notify an administrator via e-mail if a specific event occurred
- Supports the ability to notify an administrator via pager service if a specific event occurred
- Supports the ability to notify an administrator via an X windows pop (Xmessage or Xdialog etc.)
- Supports the ability to execute a program or script if a specific event occurred
- Supports the ability to pass command line or variable parameters to a script and/or program
- Supports the ability to alert an enterprise management system via SNMP greater than version1
- Support the ability to escalate events as they reach certain thresholds

- Warning mechanism should be in place in the event where large volumes of traffic overload packet-capturing capability
- Supports an acknowledgement function at the console via a pop up window or other similar mechanism if a specific event occurred
- Time out mechanism will alert console if the sensor or 'sniffer' becomes disabled

10. Countermeasure Capability

The objective of Countermeasure capability is to ensure that corrective controls can be instigated by the IDS when specific event triggered. The countermeasure capability should include the following:

- Terminate sessions when a particular event occurs
- Reroute session when a particular event occurs
- Ability to modify router ACL's with time limits
- Ability to modify the firewall rule set with time limits
- Ability to throttle back excessive bandwidth on attacks based on ICMP Source Quench Packets
- Capability to backtrack hack attempts to source reconnaissance including detecting if IP source is legitimate or spoofed

11. Maintenance and Support

The objective of maintenance and support is to provide a framework for the potential IDS vendors to benchmark their support options, skill sets required to manage and maintain the system, as well as options available for retrieving updates for the IDS product. Following is a list of maintenance and support issues that need to be addressed during the evaluation phase:

- IDS attack signature update notification can be obtained through e-mail
- IDS attack signature update notification can be obtained through postage mail with the updated attack signature software on an external medium such as disk or CD-ROM
- IDS attack signature update can be obtained via the Internet using HTTP or FTP
- IDS has a method of obtaining the attack signature updates without losing current IDS configurations
- IDS sensor should allow automated updates of OS, driver, application and management software
- IDS sensor should offer network management services
- IDS sensor should offer remote diagnostic tools
- IDS sensor should have restore/backup functionality
- IDS software does not require compiling

- IDS is easy to deploy; does not require special skills to implement
- IDS does not require special skills to maintain
- IDS attack signature database updates are provided free of charge
- Alternative sources for technical support are available
- Vendor provides 24x7 1st level support
- Signature Updates are provided on a weekly basis
- Signature Update can be both manual or automatic
- Formal training is available locally to areas where IDS is implemented

The goal of this paper is to enable the potential evaluator of IDS to form different but meaningful categories for evaluation. This can assist the evaluator to further develop a Matrix scoring system. If a scoring system is used, it should be noted that for real-time systems, importance on speed and accuracy of attack recognition, ability of the IDS to automatically react via firewall, router, SNMP etc. should be emphasized. However, for distributed systems, careful consideration should be given on the trust amongst trusted hosts. With different categories formed and Matrix scoring developed, it will allow for maximum transparency and less subjective decision.

Reference

http://www.commoncriteria.org/cem/cem.html

http://www.iatf.net/protection_profiles/intrusion.cfm

http://www.nswc.navy.mil/ITT/documents/2002_ipdps_gf.pdf

http://www.infosecuritymag.com/articles/august01/cover.shtml

http://www.nfr.com/publications/white-papers/Benchmarking-IDS-NFR.pdf (Registration with Vendor is required before site can be assessed)

http://www.nss.co.uk/ids/edition3/index.htm

(Registration with Vendor is required before this site can be assessed)

www.silicondefense.com/idwg/draft-ietf-idwg-requirements-07.txt

www.silicondefense.com/idwg/draft-ietf-idwg-beep-idxp-04.txt

http://www.ietf.org/

http://www.isi.edu/gost/cidf/

http://www.opsec.com/

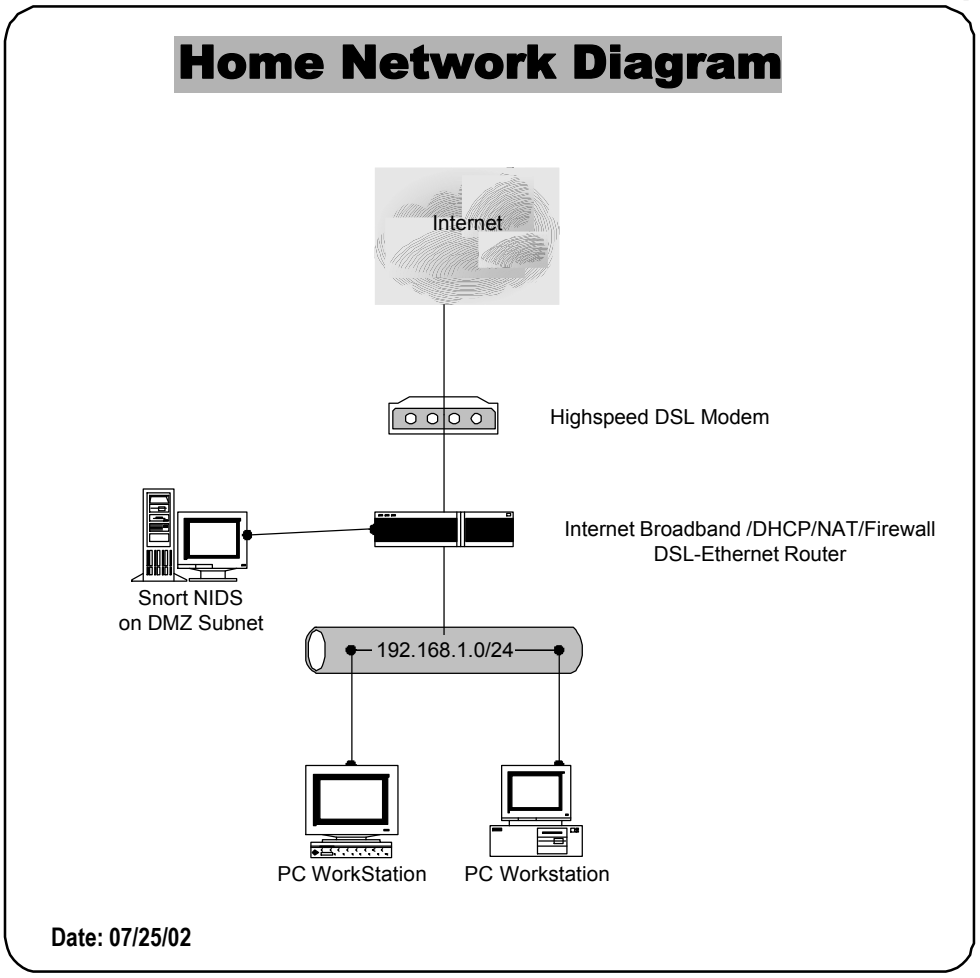
http://www.nist.gov/

http://www.intrusion.com/

("Evaluating Network Intrusion Detection Systems" 2001 white paper by Intruion.Inc. was used for research. No active link is available for this paper.)

Assignment 2 – Network Detects

Figure 1: Diagram of Home Network (Used for Detects 2 and 3)



Note: The following is a network diagram of my home network. The high speed DSL modem connects to a Linksys BEFSR41 Ethernet Router. The router is equipped with limited ACL capability and NAT's (Network Address Translation) all internal hosts to a RFC 1918 compliant address scheme. All internal hosts are assigned with a class C 192.168.1.0/24 range. The Snort IDS sensor was placed as a DMZ host and is running Snort 1.8.7b121 release, WinPcap 2.3, MySQL 3.23.40, PHP 4.1.1, PHPLot 4.4.6, ADODB 1.72, and ACID 0.96b21. Configuration of these components were achieved following the installation procedures found at http://siliconedefense.com/techsupport/winsnortacid-iis_1.8.7.htm

Detect 1 -Attempted DNS Zone Transfer

From the http://www.incidents.org/logs/Raw/2002.6.3 log , two alerts for the DNS Zone

Transfer were detected.
The following command line was used to extract the tcpdump binary file (2002.6.3) to a readable snort alert file in an ASCII format: **c:\snort\snort –r 2002.6.3 –c snort.conf**
Format [<path to snort> snort.exe <options: –r (read binary file); -c (use snort rules file)>]

[**] [1:255:5] DNS zone transfer [**]
[Classification: Attempted Information Leak] [Priority: 2]
07/03-13:29:02.774488 216.30.135.34:1099 -> 46.5.180.250:53
TCP TTL:46 TOS:0x0 ID:59822 IpLen:20 DgmLen:80 DF
AP Seq: 0x5A24D0B5 Ack: 0xACDFAF4A Win: 0x7D78 TcpLen: 32
TCP Options (3) => NOP NOP TS: 4941517 580472925
[Xref => http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0532]
[Xref => http://www.whitehats.com/info/IDS212]

[**] [1:255:5] DNS zone transfer [**]
[Classification: Attempted Information Leak] [Priority: 2]
07/03-13:52:27.734488 216.30.135.34:1107 -> 46.5.180.250:53
TCP TTL:46 TOS:0x0 ID:2616 IpLen:20 DgmLen:80 DF
AP Seq: 0xB2E57BE3 Ack: 0x4B1A9F3 Win: 0x7D78 TcpLen: 32
TCP Options (3) => NOP NOP TS: 5082012 580613435
[Xref => http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0532]
[Xref => http://www.whitehats.com/info/IDS212]

Same detect using windump (win32 version of tcpdump) which allows analyst to see more verbose raw output details.

C:\>windump -Xvn -r 2002.6.3 "host 216.30.135.34"
Format [<path to windump> windump.exe <options: -X (print in HEX and ASCII); -v (verbose output); –n (don't resolve addresses); –r (read binary file); "" (expression to match host IP)]

13:29:02.774488 216.30.135.34.1099 > 46.5.180.250.53: P [bad tcp cksum 301a!] 15
12362165:1512362193(28) ack 2900340554 win 32120 <nop,nop,timestamp 4941517 5804
72925> (DF) (ttl 46, id 59822, len 80, bad cksum 26bf!)
0x0000   4500 0050 e9ae 4000 2e06 26bf d81e 8722   E..P..@...&....."
0x0010   2e05 b4fa 044b 0035 5a24 d0b5 acdf af4a   .....K.5Z$.....J
0x0020   8018 7d78 9fdc 0000 0101 080a 004b 66cd   ..}x.........Kf.
0x0030   2299 505d 001a 1105 0000 0001 0000 0000   ".P]............
0x0040   0000 0458 5858 5803 636f 6d00 00fc 0001   ...XXXX.com.....

13:52:27.734488 216.30.135.34.1107 > 46.5.180.250.53: P [bad tcp cksum 301a!] 30
01383907:3001383935(28) ack 78752243 win 32120 <nop,nop,timestamp 5082012 580613

```
435> (DF) (ttl 46, id 2616, len 80, bad cksum 636!)
0x0000   4500 0050 0a38 4000 2e06 0636 d81e 8722      E..P.8@....6..."
0x0010   2e05 b4fa 0453 0035 b2e5 7be3 04b1 a9f3      .....S.5..{.....
0x0020   8018 7d78 df27 0000 0101 080a 004d 8b9c      ..}x.'.......M..
0x0030   229b 753b 001a 3197 0000 0001 0000 0000      ".u;..1.........
0x0040   0000 0458 5858 5803 636f 6d00 00fc 0001      ...XXXX.com.....
```

1.1 Source of Trace

Source of this trace was taken from http://www.incidents.org/logs/Raw/2002.6.3. These log files contain tcpdump binary files produced by a snort rules set.

1.2 Detect Generated By

Detect was generated by snort 1.8 . The snort rule that this detect was generated from this alert.

```
Alert[1] tcp[2] $EXTERNAL_NET any[3] ->[4] $HOME_NET 53[5] (msg:"DNS zone
transfer"; flags:A+; content: "|00 00 FC|"; offset:13;
reference:cve,CAN-1999-0532; reference:arachnids,212;
classtype:attempted-recon; sid:255;  rev:6;)
```

Explanation of Snort Rule Format			
SNORT RULE HEADER	#	**Field**	**Description**
	1.	**Alert**	Snort is informed to generate an "Alerts" file and log the detect as well. This is the default config when you install snort by default.
	2.	**Tcp**	Snort analyses for four protocols for suspicious behavior - tcp, udp, icmp, and ip. This rule is defined only for TCP related traffic.
	3.	**$EXTERNAL_NET any**	This portion deals with the Source IP address and port information for a given rule. The word "any" can be used as a wildcard for each address and port. If $EXTERNAL_NET is not defined it can generally refer to all/any external IP addresses.
	4.	**->**	This indicates the direction of the traffic for which the snort rule shall apply. Greater than symbol means that the ip address and port from the left hand side is the source address and ip address and port on the right indicates the destination address.
	5.	**$HOME_NET 53**	This portion deals with the destination IP address and port number for a specific rule. $Home_Net is a user defined variable to refer to a defined host/network. The destination is focused on a port defined to Domain Name Services (DNS).
SNORT RULE OPTIONS	**6.**	**(msg:"DNS zone transfer"; flags:A+; content: "|00 00 FC|"; offset:13; reference:cve,CAN-1999-0532; reference:arachnids,212; classtype:attempted-recon; sid:255; rev:6;)**	This portion is the Rule Options of Snort: **Msg:"DNS zone transfer"** -defines message to rule i.e DNS zone transfer **Flags: A+; content: "|00 00 FC|"; offset:13;** - defines packet attributes. In this rule, flags should have the TCP ACK flag and other TCP flags set. Content from the 13th byte offset from the packet payload is screened for the HEX characters 00 00 FC to flag an attempted zone transfer. **Reference: cve etc..** –defines associated CVE/arachnids/cert public advisories for more details. Vulnerability has also been classified with well-known open forum.

1.3 Probability The Source Address Was Spoofed

This source IP address "216.30.135.34" in all probability was not spoofed. The nature of this detect requires that a full 3 way duplex TCP handshake to be completed in order to function properly. DNS zone transfers require that the source address be legitimate in order to receive the reconnaissance information to a "real" source IP address. Evidence from Arachnids supports our conclusion in stating: "The packet that caused this event is normally a part of an established TCP session, indicating that the source IP address has not been spoofed." (http://www.whitehats.com/info/IDS212)

1.4 Description of The Attack

The nature of a zone transfer is to obtain information on a target host. In order to understand the attack, we must first understand what a Zone is. Domains are broken into "zones" for which individual DNS servers are responsible. A domain represents the entire set of names / machines that are contained under an organizational domain name. For example all domain names ending with ".com" are part of the "com" domain. Entire Zones are transferred from a primary DNS server to secondary DNS servers through Zone Transfers. Other than the legitimate zone transfer that occur between a primary and secondary DNS server, detects of zone transfers are merely attackers who are gathering information about your domain and profile your Internet footprint or presence. Many DNS servers are mis-configured thus allow attackers to use resolvers like nslookup. A resolver is library routines that create queries and send them across a network to a name server. Attackers use tools like nslookup to attempt zone transfers on these name servers. Information obtained from a zone transfer range from all types of applications from web servers, mail servers, ftp servers, to gateways and other DNS servers. As many corporations have a tendency of naming hosts by their function (i.e. fw.companyabc.com and ftp.companyabc.com) attackers find this information very useful in targeting hosts like ftp servers and corporate firewalls.

1.5 Attack Mechanism

Zone transfers work by TCP, as they require reliable means of transmission in order to download/exchange zone info for primary/secondary DNS servers. The alert is triggered by any external network from any port number that has a destination to the protected home network of a destination with port 53 with the TCP protocol set. The alert functions by looking only at the 13th byte offset of the payload for the hex string 00 00 FC. A zone transfer is the QTYPE or query type of 252 or Hex FC.
An attacker performs a zone transfer by following these steps:

1) Start a windows/unix command shell and types nslookup
2) Use the "server" command to change the default DNS server to the authorative DNS server of the domain we want to query
3) Use the command "set type=any" to allow all types of records to show
4) Type "ls –d companyabc.com." to list all the records for the domain. The "." At

the end signifies we are looking for fully qualified domain names.
Because of the IP of the source IP address is different from the destination, we can infer that this detect is not a false positive as primary slave DNS servers are typically on the same class subnet. In addition the TTL value is 46, a number that indicates that the host was multiple hops away from the destination DNS server. A low ttl value size could mean Linux 2.2.x kernel (default 64) and a default window size of 32120 also supports this (see http://project.honeynet.org/papers/finger/traces.txt). This would also explain the checksum errors that this signature included. Linux 2.2.x kernel was susceptible to this and we can validate this from the site below.
http://www.uwsg.iu.edu/hypermail/linux/kernel/0111.1/0197.html

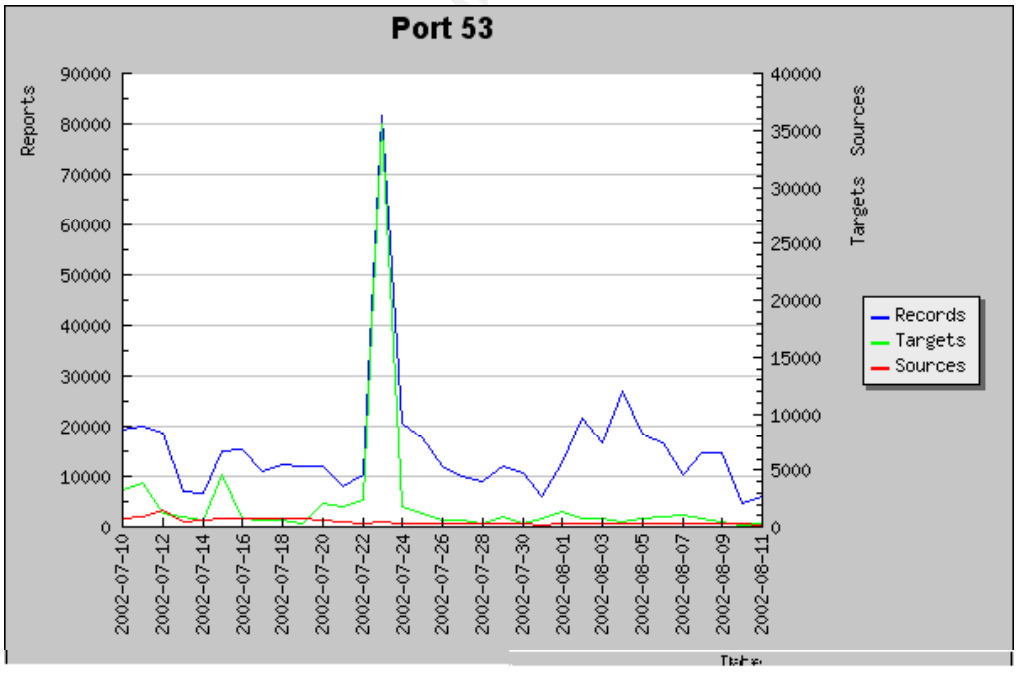
1.6 Correlations

Following sources note correlations to the nature of this detect

1) ARIN:

Jump Point Communications, Inc. (NETBLK-JUMP-BLK-3)
7218 McNeil Drive, Suite 310
Austin, TX 78729
US
Netname: JUMP-BLK-3
Netblock: 216.30.0.0 - 216.30.143.255
Maintainer: JUMP

2) http://isc.incidents.org/port_details.html?port=53 The following chart represents port 53 activity to reported records, charts, and sources from July-August 2002.



3) DNS appears to be the top 14 from the http://isc.incidents.org/port_report.html

with respect to overall reports as of August 2002
All registered to Jump.net's partial Class B address listing and reported to incident.org.
No specific address from our source host was registered but this is a prudent step in order to associate any previous threats from that source.

Source	Sources	Targets	Reports
216.030.008/24	2	12	28
216.030.011/24	1	4	6
216.030.019/24	1	1	2
216.030.025/24	1	1	1
216.030.039/24	2	56	159
216.030.040/24	1	5	6
216.030.046/24	1	1	2
216.030.067/24	1	1	1
216.030.078/24	1	1	1
216.030.097/24	2	156	166
216.030.100/24	1	2	3
216.030.104/24	2	25	39
216.030.108/24	1	1	1
216.030.116/24	1	1	1
216.030.120/24	2	4	5
216.030.134/24	1	2	7
216.030.140/24	1	2	8

4) a) http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0532 -CVE
b) http://www.whitehats.com/info/IDS212 - Arachnids -whithats.com
c) http://www.iss.net/security_center/advice/Intrusions/2000401/default.htm -ISS

1.7 Evidence of Active Targeting

No evidence of active targeting is apparent. The nature of zone transfers is a sign of not active targeting but reconnaissance. The next logical step after an attacker

gathers necessary information about all the associated zones mapping Ip addresses to hostnames she/he might then possibly move in the next phase of active targeting a specific host or service. DNS interrogation is a sign not actively associated to target a single host but more likely an organization and associated domains that are registered to it.

1.8 Severity

Severity is determined by using the following formula: Each metric is graded on a five-point scale, with five being the highest and one being the lowest.	
Severity =	**(Criticality + Lethality) - (System + Network Countermeasures)**
Criticality: 5	DNS servers are critical, as they always must be exposed on the Internet.
Lethality: 1	This detect is not lethal as it just tries to mimic the legitimate feature of a secondary DNS server.
System Countermeasures: 4	This is unknown so we must assume that no system countermeasures are available. I.e. firewall, latest patch of dns software (i.e. bind, MSDNS) or how well the host is hardened.
Network Countermeasures: 4	This is unknown, we cannot validate the position of the DNS server. We can confirm that the there is IDS as we were able to positively identify this detect.
The Severity in this case is: -2	

1.9 Defensive Recommendation

This detect only showed an attempted DNS Zone transfer. We do not have more details and should assume worst-case scenario and thus recommend Industry best practice. Therefore the following should be implemented.

- DNS Zone Transfers should be restricted to only authorized DNS servers
- DNS servers implement a split DNS strategy thereby using only public IP to Hostname mappings on the Internet side.
- Do not use HINFO to volunteer unnecessary information (i.e. O/S info) to the public domain nor name hosts by their function as in the example fw.companyabc.com.
- Configure the external firewall or packet filter to deny all unauthorized inbound TCP port 53 connections since name lookups use UDP port 53.

Preventing zone transfers will increase the difficulty of attackers trying to footprint or

get a detailed map of your network.

1.10 Multiple Choice Test Question

A DNS Zone Transfer detect is a sign of:
a) Legitimate traffic replicating between a Secondary DNS server and a Primary DNS server.
b) An administrator downloading a copy of the DNS server config file.
c) An Attacker using a resolver like "nslookup" to query a DNS server.
d) Both A &C
Answer: D

Detect 2 – FTP WU-FTP File Completion Attempt

ID	< Signature >	< Timestamp >	< Source Address >	< Dest. Address >	< Layer 4 Proto >
#0-(1-2222)	[CVE] [bugtraq] FTP wu-ftp file completion attempt [	2002-06-24 16:31:47	24.150.32.131:1144	192.168.1.100:21	TCP
#1-(1-2221)	[CVE] [bugtraq] FTP wu-ftp file completion attempt [	2002-06-24 16:31:47	24.150.32.131:1144	192.168.1.100:21	TCP
#2-(1-2220)	[CVE] [bugtraq] FTP wu-ftp file completion attempt [	2002-06-24 16:31:47	24.150.32.131:1144	192.168.1.100:21	TCP
	Continued….				

Meta	ID # Time Triggered Signature 1 - 2221 2002-06-24 16:31:47 [CVE] [bugtraq] FTP wu-ftp file completion attempt [
	name interface filter SNORTBOX:DevicePacket_{00EA8953-47B6-4C47-9102-6E8629B84BFF} DevicePacket_{00EA8953-47B6-4C47-9102-6E8629B84BFF} *none*
	none

IP	source addr dest addr Ver Hdr Len TOS length ID flags offset TTL chksum 24.150.32.131 192.168.1.100 4 5 0 1500 12491 0 0 117 5164
	Source Name Dest. Name d150-32-131.home.cgocable.net snortbox
	none

TCP	source port dest port R 1 R 0 U R G A C K P S H R S T S Y N F I N seq # ack offset res window urp chksum 1144 21 X 3793851945 2656014108 5 0 17520 0 51115
	none

```
Payload length = 1460
Excerpt ~ + [ bolded and underlined for clarity.

200 : 80 7E 82 78 53 79 06 FB D3 D0 D7 91 29 88 1B 9D   .~.xSy.....)...
210 : DA 40 1F F7 C0 13 A2 11 CE 59 4B 02 1E 67 93 A8   .@.......YK..g..
220 : DD 48 33 00 6D DE B6 7E B7 AE EB A0 69 AD 31 D8   .H3.m..~....i.1.
---removed for clarity----
3d0 : BF 57 77 A0 19 00 99 1C EE EC 24 ED F7 5B 99 4C   .Ww.......$..[.L
---removed for clarity----
420 : 97 6E 19 02 E4 1A B7 18 BE 6D 91 51 5B 61 59 7E   .n.......m.Q[aY~
430 : E2 53 56 A8 8C 10 2A 15 3E 4B AD 8F A3 FB A4 E6   .SV...*.>K......
440 : FF AF 30 E1 60 4D 0F C3 84 23 0C 93 7B 18 5B 22   ..0.`M...#..{.["
450 : D9 07 DC DE 07 B4 15 B4 79 3F 70 DB 0F DA EE 69   ........y?p....i
---removed for clarity----
520 : A3 C3 6F 9C 1D 8B 10 E1 D6 CC CD 5B AE 74 87 B9   ..o........[.t..
---removed for clarity----
580 : CE 06 92 5B 10 BA 15 84 6D 32 F4 44 60 30 F4 0F   ...[....m2.D`0..
```

2.1 Source of Trace

Home Network See Figure:1 for Layout

2.2 Detect Generated By

Detect was generated by snort 1.8 . The snort rule that this detect was generated from this alert.

```
Alert[1] tcp[2] $EXTERNAL_NET any[3] ->[4] $HOME_NET 21[5] (msg:"FTP wu-ftp file
completion attempt ["; flags:A+; content:"~"; content:"[";
reference:cve,CAN-2001-0886; reference:bugtraq,3581; classtype:misc-
attack; sid:1377;[6]
```

Explanation of Snort Rule Format			
SNORT RULE HEADER	**#**	**Field**	**Description**
	1.	**Alert**	Snort is informed to generate an "Alerts" file and log the detect as well. This is the default config when you install snort by default.
	2.	**Tcp**	Snort analyses for four protocols for suspicious behavior - tcp, udp, icmp, and ip. This rule is defined only for TCP related traffic.
	3.	**$EXTERNAL_NET any**	This portion deals with the Source IP address and port information for a given rule. The word "any" can be used as a wildcard for each address and port. If $EXTERNAL_NET is not defined it can generally refer to all/any external IP addresses.
	4.	**->**	This indicates the direction of the traffic for which the snort rule shall apply. Greater than symbol means that the ip address and port from the left hand side is the source address and ip address and port on the right indicates the destination address.
	5.	**$HOME_NET 21**	This portion deals with the destination IP address and port number for a specific rule. $Home_Net is a user defined variable to refer to a defined host/network. The destination is focused on a port defined to the File Transfer Protocol (FTP).

SNORT RULE OPTIONS	**6.**	**(msg:"FTP wu-ftp file completion attempt ["; flags:A+; content:"~"; content:"["; reference:cve,CAN-2001-0886; reference:bugtraq,3581; classtype:misc-attack; sid:1377;**	This portion is the Rule Options of Snort: **Msg:" FTP wu-ftp file completion attempt"** -defines message to rule i.e **Flags: A+; content: :"~"; content:"[";** - defines packet attributes. In this rule, flags should have the TCP ACK flag and possibly others set. We are also expected to see a left bracket and tilde symbol in the content of the packet payload to signify a wu-ftp file completion attempt. **Reference: cve etc..** –defines associated CVE/bugtraq public referenced advisories for more details. Vulnerability has also been classified with well-known open forum.

2.3 Probability The Source Address Was Spoofed

There is a low probability that this detect was spoofed. To successfully perform this exploit, it becomes necessary to receive responses from your target, and participate in a complete TCP 3 way handshake and session. Spoofed IP addresses have limited functionality and are usually associated to attacks where Denial of Service is involved and the attacker is not expecting a response to his packets sent to the target host. This IP address was not spoofed.

2.4 Description of The Attack

Wu-Ftpd is a ftp server developed and maintained by Washington University. Wu-ftp is a server known to be supported on linux/unix based operating systems. The attack to vulnerable FTP servers allows for clients to organize files for ftp actions based on "file globbing" patterns. Globbing is used to expand special characters in a wildcard name, or the act of so doing (the action is also called `globbing'). The wu-ftp server implementation of file globbing contains a heap corruption vulnerability that can allow an attacker to execute arbitrary code successfully on the ftp server remotely. A heap corruption is similar to a buffer overflow condition in that the end result is the same, the memory where the data is stored is not checked and can overwrite the pointers or return addresses to execute arbitrary code. If the ftp server allows for anonymous ftp access, anyone can execute this overflow, else only an authorized user can attempt this exploit. The most notable CVE related to this detect is CAN-2001-0886.

2.5 Attack Mechanism

The attack was first released to public around the November/December 2001 timeframe. The attack works by first having anonymous or authorized access to a vulnerable wu-ftp server. A user then executes a file globbing pattern. During this action the data (globbed filenames) is stored on the heap using the malloc() function. Malloc is just short for memory allocation. This exploit utilizes the file glob function in that it does not perform checking when the function is processing the file glob pattern. This means that it may be possible to have an arbitrary word in memory overwritten with an arbitrary value. This can lead to the execution of arbitrary code because the function pointers or return memory addresses are overwritten. A role that should be performed by the glob function call but isn't. To be successful, the attacker must craft a malicious malloc header

containing the target address and the proper value in the right memory address in the heap so it can be executed. The FTP command must be one that will not set an error variable. This is why Snort is looking for special globbing characters like ~,{ or[as these are specific characters that will not set an error variable When the server attempts to free the memory used to store the globbed filenames, the target word in memory will be overwritten. The attacker will also ensure that the proper malicious shell code is appended and processed by the server.

2.6 Correlations

1) Time Profile of Alerts by Source 24.150.32.131

Time	#Alerts	Alerts
06/24/2002 9:00:00 - 9:59:59	1	
06/24/2002 10:00:00 - 10:59:59	0	
06/24/2002 11:00:00 - 11:59:59	17	
06/24/2002 12:00:00 - 12:59:59	36	
06/24/2002 13:00:00 - 13:59:59	28	
06/24/2002 14:00:00 - 14:59:59	0	
06/24/2002 15:00:00 - 15:59:59	12	
06/24/2002 16:00:00 - 16:59:59	15	

Note: Total 108 attempts in 8 hours. Produced by ACID Useful in exploring the patterns of the detects occurrence.

2) Correlation to Online Vulnerability Databases.

- http://aris.securityfocus.com/alerts/wuftpd/011128-Alert-wuftpd.pdf
- http://online.securityfocus.com/bid/3581/info/
- http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2001-0886

3) Arin Search useful in determining source IP origin.
Cogeco Cable Solutions (NETBLK-CGOC-HALA1-1)

950 Syscon Drive
Burlington, ON L7R 4S6
CA
Netname: CGOC-HALA1-1
Netblock: 24.150.32.0 - 24.150.47.255
Coordinator:
Cogeco Cable (IS7-ORG-ARIN) ipservices@cogeco.net
905-333-7055

4) According to Security Focus Research WU-ftp file globbing exploit was the top 4th for the first quarter of 2002.
http://www.securityfocus.com/corporate/research/top10attacks_q1_2002.shtml

2.7 Evidence of Active Targeting

There does not seem to be evidence of active targeting. It is most likely a case of a false positive brought on by the vague snort signature patterns that is attempting to flag this exploit. This detect was most probably triggered as part of a large binary file download. Special characters detected in the payload ([and ~) were similar to the wu-ftp file globbing heap corruption pattern detected. This was not an active target. It is part of a normal file transfer. We can corroborate this by the fact that our ACID/Snort did not detect the source IP address with any other signature from a 2-month period. We might become suspicious if we noted any type of reconnaissance scans that might have occurred prior to this detect. The nature of this detect would probably be more of a targeted attack as this detect is an attack signature and would probably be more of a concern had my FTP server was running WU-FTP.

2.8 Severity

Severity is determined by using the following formula: Each metric is graded on a five-point scale, with five being the highest and one being the lowest.	
Severity =	**(Criticality + Lethality) - (System + Network Countermeasures)**
Criticality: 2	The ftp server did not allow anonymous access and not a critical box with critical information.
Lethality: 5	Heap corruptions are the most lethal. As this particular exploit resulted in an attacker running arbitrary code remotely suggests that availability, confidentiality and integrity are lost.

System Countermeasures: 5	FTP server is running patched and hardened IIS Server according to Microsoft security guidelines. WU-ftp is not installed on network. No anonymous access is allowed.
Network Countermeasures: 4	External router only enables authorized external ftp access within an as per need basis. All other times inbound ftp access is blocked by the router using ACL's. NIDS in place and detected with ACID.
The Severity in this case is: -2	

2.9 Defensive Recommendation

If you are running Washington University FTP server or any application that uses its source:

- Limit access to the wu-ftpd service by allowing only authorized users and limiting access from authorized hosts (specific IP's and/or networks).
- Disable anonymous FTP access, allowing only authorized users.
- Disable the FTP service entirely, until all patches have been installed.
- Installing Host Based IDS on Internet FTP servers can also detect/intercept buffer and heap overflows and protect the kernels from such attacks. Newer HIDS operate by intercepting all kernel call and proxy requests thus can prevent these types of activities.
- Subscribe to security mailing lists to learn about exploits when they become public.

If you are not running WU-FTP in your environment, you may wish to tune your IDS and remove this detect, as it is specific to an application and can eliminate false positives for your organization.

2.10 Multiple Choice Test Question

In November/December 2001 "wu-ftp servers" reported to contain a heap corruption with its implementation of:
a) file globbing
b) binary file transfers
c) TCP/IP stack
d) Microsoft IIS
Answer: A

Detect 3 – Scan Socks Proxy Attempt

[**] SCAN SOCKS Proxy attempt [**]
06/01-20:02:43.084999 68.68.242.72:3191 -> 192.168.1.100:1080
TCP TTL:102 TOS:0x0 ID:50386 IpLen:20 DgmLen:48 DF
******S* Seq: 0x117CA1E1 Ack: 0x0 Win: 0x2000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK
=+
=+=+=+=+=+

[**] SCAN SOCKS Proxy attempt [**]
06/01-20:02:43.847621 68.68.242.72:3191 -> 192.168.1.100:1080
TCP TTL:102 TOS:0x0 ID:64210 IpLen:20 DgmLen:48 DF
******S* Seq: 0x117CA1E1 Ack: 0x0 Win: 0x2000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK
=+
=+=+=+=+=+

[**] SCAN SOCKS Proxy attempt [**]
06/01-20:02:44.604425 68.68.242.72:3191 -> 192.168.1.100:1080
TCP TTL:102 TOS:0x0 ID:18131 IpLen:20 DgmLen:48 DF
******S* Seq: 0x117CA1E1 Ack: 0x0 Win: 0x2000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK
=+
=+=+=+=+=+

[**] SCAN SOCKS Proxy attempt [**]
06/01-20:02:45.328192 68.68.242.72:3191 -> 192.168.1.100:1080
TCP TTL:102 TOS:0x0 ID:31699 IpLen:20 DgmLen:48 DF
******S* Seq: 0x117CA1E1 Ack: 0x0 Win: 0x2000 TcpLen: 28
TCP Options (4) => MSS: 1460 NOP NOP SackOK
=+
=+=+=+=+=+

C:\>windump -Xvn -r 2002.6.3 "host 68.68.242.72"
Format [<path to windump> windump.exe <options: -X (print in HEX and ASCII); -v (verbose output); –n (don't resolve addresses); –r (read binary file); "" (expression to match host IP)]
20:02:43.084999 68.68.242.72.3191 > 192.168.1.100.1080: **S** [tcp sum ok] 293380577:293380577(0) win 8192 <mss 1460,nop,nop,sackOK> (DF) (ttl 102, id 50386, len 48)
0x0000 4500 0030 c4d2 4000 6606 575c 4444 f248   E..0..@.f.W\DD.H
0x0010 c0a8 0164 0c77 0438 117c a1e1 0000 0000   ...d.w.8.|......
0x0020 7002 2000 a679 0000 0204 05b4 0101 0402   p....y..........

20:02:43.085137 192.168.1.100.1080 > 68.68.242.72.3191: **R** [tcp sum ok] 0:0(0) **ack**

293380578 win 0 (ttl 128, id 2622, len 40)
0x0000 4500 0028 0a3e 0000 8006 37f9 c0a8 0164 E..(.>....7....d
0x0010 4444 f248 0438 0c77 0000 0000 117c a1e2 DD.H.8.w.....|..
0x0020 5014 0000 f329 0000 P....)..

20:02:43.847621 68.68.242.72.3191 > 192.168.1.100.1080: **S** [tcp sum ok] 293380577:293380577(0) win 8192 <mss 1460,nop,nop,sackOK> (DF) (ttl 102, id 64210, len 48)
0x0000 4500 0030 fad2 4000 6606 215c 4444 f248 E..0..@.f.!\DD.H
0x0010 c0a8 0164 0c77 0438 117c a1e1 0000 0000 ...d.w.8.|......
0x0020 7002 2000 a679 0000 0204 05b4 0101 0402 p....y.........

20:02:43.847811 192.168.1.100.1080 > 68.68.242.72.3191: **R** [tcp sum ok] 0:0(0) **ack** 1 win 0 (ttl 128, id 2623, len 40)
0x0000 4500 0028 0a3f 0000 8006 37f8 c0a8 0164 E..(.?....7....d
0x0010 4444 f248 0438 0c77 0000 0000 117c a1e2 DD.H.8.w.....|..
0x0020 5014 0000 f329 0000 P....)..

20:02:44.604425 68.68.242.72.3191 > 192.168.1.100.1080: **S** [tcp sum ok] 293380577:293380577(0) win 8192 <mss 1460,nop,nop,sackOK> (DF) (ttl 102, id 18131, len 48)
0x0000 4500 0030 46d3 4000 6606 d55b 4444 f248 E..0F.@.f..[DD.H
0x0010 c0a8 0164 0c77 0438 117c a1e1 0000 0000 ...d.w.8.|......
0x0020 7002 2000 a679 0000 0204 05b4 0101 0402 p....y.........

20:02:44.604590 192.168.1.100.1080 > 68.68.242.72.3191: **R** [tcp sum ok] 0:0(0) **ack** 1 win 0 (ttl 128, id 2624, len 40)
0x0000 4500 0028 0a40 0000 8006 37f7 c0a8 0164 E..(.@....7....d
0x0010 4444 f248 0438 0c77 0000 0000 117c a1e2 DD.H.8.w.....|..
0x0020 5014 0000 f329 0000 P....)..

20:02:45.328192 68.68.242.72.3191 > 192.168.1.100.1080: **S** [tcp sum ok] 293380577:293380577(0) win 8192 <mss 1460,nop,nop,sackOK> (DF) (ttl 102, id 31699, len 48)
0x0000 4500 0030 7bd3 4000 6606 a05b 4444 f248 E..0{.@.f..[DD.H
0x0010 c0a8 0164 0c77 0438 117c a1e1 0000 0000 ...d.w.8.|......
0x0020 7002 2000 a679 0000 0204 05b4 0101 0402 p....y.........

20:02:45.328352 192.168.1.100.1080 > 68.68.242.72.3191: **R** [tcp sum ok] 0:0(0) **ack** 1 win 0 (ttl 128, id 2625, len 40)
0x0000 4500 0028 0a41 0000 8006 37f6 c0a8 0164 E..(.A....7....d
0x0010 4444 f248 0438 0c77 0000 0000 117c a1e2 DD.H.8.w.....|..
0x0020 5014 0000 f329 0000 P....)..

3.1 Source of Trace

Home Network See Figure:1 for Layout

3.2 Detect Generated By

Detect was generated by snort 1.8 . The snort rule that this detect was generated from this alert.

```
Alert[1] Tcp[2] $EXTERNAL_NET any[3] ->[4] $HOME_NET 1080[5] (msg:"SCAN SOCKS
Proxy attempt"; flags:S; reference:url,help.undernet.org/proxyscan/;
classtype:attempted-recon; sid:615; rev:3;)[6]
```


Explanation of Snort Rule Format			
SNORT RULE HEADER	**#**	**Field**	**Description**
	1.	**Alert**	Snort is informed to generate an "Alerts" file and log the detect as well. This is the default config when you install snort by default.
	2.	**Tcp**	Snort analyses for four protocols for suspicious behavior - tcp, udp, icmp, and ip. This rule is defined only for TCP related traffic.
	3.	**$EXTERNAL_NET any**	This portion deals with the Source IP address and port information for a given rule. The word "any" can be used as a wildcard for each address and port. If $EXTERNAL_NET is not defined it can generally refer to all/any external IP addresses.
	4.	**->**	This indicates the direction of the traffic for which the snort rule shall apply. Greater than symbol means that the ip address and port from the left hand side is the source address and ip address and port on the right indicates the destination address.
	5.	**$HOME_NET 1080**	This portion deals with the destination IP address and port number for a specific rule. $Home_Net is a user defined variable to refer to a defined host/network. The destination is focused on a port defined to Socks proxy ports.
SNORT RULE OPTIONS	**6.**	**(msg:"SCAN SOCKS Proxy attempt"; flags:S; reference:url,help.undernet.org /proxyscan/; classtype:attempted-recon; sid:615; rev:3;)**	This portion is the Rule Options of Snort: **Msg:" SCAN SOCKS Proxy attempt"** -defines message to rule. This is a scan for open socks proxies. **Flags: S;** - defines packet attributes. In this rule, TCP flags should have the TCP SYN flag set **Reference: cve etc..** –defines associated web site for more information associated to detect signature.

3.3 Probability The Source Address Was Spoofed

The source IP 68.68.242.72 was probably not spoofed for the following reasons:

- http://www.adelphia.com the registered domain from the source IP has been identified as a cable modem service provider. Many reconnaissance scans are known to stem from such environments.

- The target port of this detect is 1080. Port 1080 can be associated to socks, subseven 2.2, and Winhole. SubSeven 2.2 and Winhole are also known Trojans that an attacker uses to control the target's host and thereby requires that traffic be directed to the true source in order to function.
- Spoofed IP addresses would not benefit the attacker, as spoofed IP's would be directed to the spoofed IP and not the attacker IP.

20:02:43.084999 68.68.242.72.3191 > 192.168.1.100.1080: **S** [tcp sum ok] 293380577:293380577(0) win 8192 <mss 1460,nop,nop,sackOK> (DF) (ttl 102, id 50386, len 48)
20:02:43.085137 192.168.1.100.1080 > 68.68.242.72.3191: **R** [tcp sum ok] 0:0(0) **ack** 293380578 win 0 (ttl 128, id 2622, len 40)

In our example notice the bold type, the source host sent was a SYN to port 1080 and our network returned a rst + ack meaning that the port requested was closed. An attacker using a spoofed IP would not receive this packet unless he was using a sniffer on the subnet of the spoofed IP address.

3.4 Description of The Attack

The detect is a stimulus for an active TCP port 1080. The nature of the detect is reconnaissance as this detect is the first step before an attack. Port 1080 is significant as there are many associated legitimate and illegitimate applications. Identifying an open port to an attacker could mean Socks Proxy, Wingate Proxy, SubSeven version 2.2 (remote control malware) or Winhole (a Trojan version of Wingate that installs silently on the target host). Socks is a popular protocol for targeting and tunneling traffic through a firewall. Socks by nature allow many computers behind a computer to access the Internet without being connected. Attackers may be looking for misconfigured proxy servers to bounce their traffic and tunnel though the socks proxy to another socks proxy to stem an attack or other types of malicious behavior.

3.5 Attack Mechanism

The mechanism of the scan proxy scan attempt is by initiating a 3 way TCP handshake with a target host on port 1080. If the service is listening the server will reply with a SYN + ACK to the source host. If the proxy service is not listening on the host, the server will send a RST +ACK to the source. If an attacker determines that a service is listening she/he may begin to attampt to exploit weaknesses attributed to a poorly configured proxy server, default passwords, or attempt to exploit previously installed Trojans like Winhole and Subseven 2.2.

3.6 Correlations

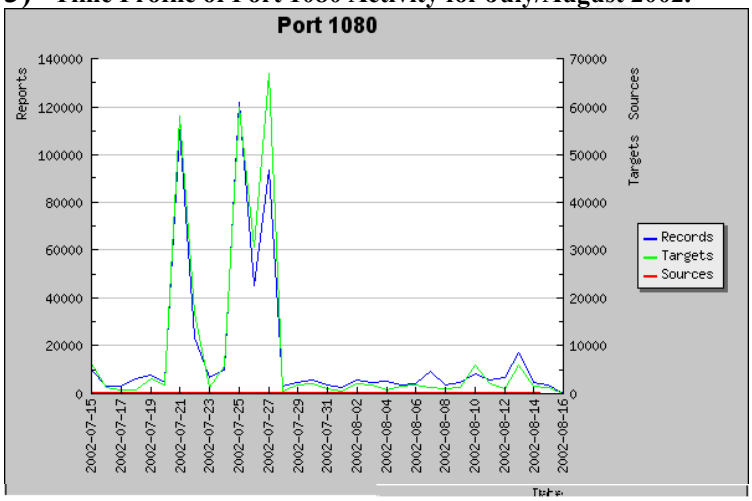
1) a) ARIN Registration of Source IP
Adelphia Cable Communications (NETBLK-ADELPHIA-CABLE-4)

Main at Water Street
Coudersport, PA 16915
US
Netname: ADELPHIA-CABLE-4
Netblock: 68.64.0.0 - 68.71.255.255
Maintainer: ADEL
Coordinator:
Hostmaster, Adelphia (AH102-ARIN) ipadmin@adelphia.net
814.274.0638 (FAX) 814.274.8457

2) The Source IP has been reported prior with the Incident Storm Center. We can see from this report that there have been 48 reports to ISC and the source has been privy to 38 targets.
http://isc.incidents.org/source_report.html?subnet=068.068.242.072

Source	Sources	Targets	Reports
068.068.242.072/32	1	38	48

3) Time Profile of Port 1080 Activity for July/August 2002.



Source: http://isc.incidents.org/port_details.html?port=1080

4) Proxy Server Vulnerabilities
http://www.iss.net/security_center/static/5373.php
http://www.iss.net/security_center/static/1849.php
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0290
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0291
http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0441

3.7 Evidence of Active Targeting

There is no direct evidence to support active targeting. Our ACID IDS logs only showed one instance of proxy scanning from a one-month period. This would probably conclude that this scan was part of a larger scan that included many more subnets performing bulk scans to find active proxy servers or compromised servers running SubSeven or Winhole. Our correlation research (see Section 3.6 #2) support our theory as the source IP was determined to be a registered attacker meaning other

network admins have experienced similar scanning/probing in the past.

3.8 Severity

Severity is determined by using the following formula: Each metric is graded on a five-point scale, with five being the highest and one being the lowest.	
Severity =	**(Criticality + Lethality) - (System + Network Countermeasures)**
Criticality: 5	The device is a Firewall/Router and is critical to the home network.
Lethality: 2	This detect is not lethal by nature. It is a recon scan that does not have any malicious behavior associated to it.
System Countermeasures: 5	No services are open on the home network. All hosts have the latest patches installed and are hardened with Industry best practice. Anti-virus software is installed on all hosts behind firewall.
Network Countermeasures: 5	Firewall/router block all inbound traffic, no services are running. IDS is in place with ACID to detect and report on network detects.
The Severity in this case is: -3	

3.9 Defensive Recommendation

Appropriate recommendations for Proxy servers include:

- Firewall policy to ensure that anything that is not explicitly permitted is denied
- If proxy services are installed that they are configured properly as per recommendations and fixes provided by http://help.undernet.org/proxyscan/. Running free online vulnerability tools like https://grc.com/x/ne.dll?bh0bkyd2 and http://scan.sygatetech.com/pretcpscan.html is a good practice to detect changes in your environment.
- Always ensure that appropriate security patches are installed in a timely fashion.
- Antivirus software installed on all servers with latest definition files to ensure that Trojan like SubSeven and Winhole are detected.

3.10 Multiple Choice Test Question

"Scan Socks Proxy Attempt" Snort Signature looks for the following TCP Flags
a) msg:"SCAN SOCKS Proxy attempt"; flags:A+
b) msg:"SCAN SOCKS Proxy attempt"; flags:A

c) msg:"SCAN SOCKS Proxy attempt"; flags:S
d) msg:"SCAN SOCKS Proxy attempt"; flags:U+
Answer: a

Assignment 3: Analyze This

1. Executive Summary

SANS has requested an analysis and review of an IDS implementation at the University of University of Maryland Baltimore County (known hereafter as "the University"). The University's IDS implementation is based on a freeware open source solution called Snort. The purpose of our review was multifaceted. Our objective was to identify potential vulnerabilities; exposures, network issues, as well as profile likely compromised internal hosts. The following report provides a comprehensive explanation of our analysis and findings as well as our recommendations for the corrective action required in each case. Where applicable we have also included diagrams to explain network traffic relationships and supporting external correlations to support our analysis. It is our assessment that University networks have the following issues.

- Default implementation of Snort IDS with little or no customization to the rule set. This is attributed to multiple false positive signatures reviewed below.
- Evidence of compromised internal hosts that require formal computer forensic investigations
- Prevalent use of Peer to Peer applications that are insecure and can cause network degradation for legitimate university application. More and more worms and buffer overflows have been identified in this space.
- Identified several external source addresses with registration info that should be further investigated.

Overall we recommend that the University consider the following:

- Enforce an approved information security policy to govern what traffic is allowed deny all unless explicitly permitted.
- Ensure all server software are hardened and patches applied in a timely fashion
- Investigate compromised hosts in a timely manner. Ensure that you have trained personal in computer forensics to secure evidence.
- Tweak the IDS to silence known traffic signatures and known traffic patterns (i.e. SMB Name Wildcard, ICMP Echo Request L3retriever Ping, and connect to 515 from inside) are examples of non malicious detects that could be muted.
- Define internal hosts variable to "var HOME_NET 130.85.0.0/16" in the snort.conf file. This will alleviate many false positives and better utilize time of IDS administrator reviewing detects that are triggered as part of expected traffic behavior. In addition poorly configured IDS rules will mount run-time costs as log files can grow large and add unnecessary management and maintenance costs to support the IDS solution.
- Perform security audits regularly and report to senior executive teams on critical

security issues.

- Investigate the need to secure Peer to Peer and IM services. Applications like Kazaa and MSN IM are becoming extremely popular and is something the University should take a stand on as it can lead to information leakage and waste critical network bandwidth better suited for educational research rather than recreational use.

Please note that as any report only captures a point in time, we highly recommend that the recommended preventative, and detective controls discussed in this report be implemented and tested in a timely fashion to ensure that security processes are continual and effective. An IDS is ineffective unless it is directed and controlled by an Information Security Policy that dictates how it functions. There is a saying that a fool with a tool is a fool and a fool with a bigger tool is a bigger fool, the same applies to an IDS implementations, it is only as good as the policies, people and process that manage and govern over the technology.

2. Scope and Methodology

The scope of the engagement was based on the analysis and review of five consecutive days of IDS log file sets. The IDS logs reviewed were Alert, Scan and Oos log files. Alert files are ASCII log files containing possible signature matches as identified by the snort rule set. Scan files are also ASCII log files that contain network reconnaissance scans that are defined by the Snort IDS. Oos files are" Out of Spec" ASCII log files that detail irregular packet construction and possibly evidence of packet crafting across a network. The following files were reviewed in this report. MD5 Hashes are also used to verify the integrity of the log files as obtained from the www.incidents.org/logs University log repository.

Date	Alert	Scan	OOS
June 5 2002	Alert.02.0605.gz e8eb826a1c2b584a660f1987ee8e05ef	Scans.02.0605.gz d91a975499578220464e89bcee1ecc0f	Oos_Jun.5.2002.gz e40d012d3bf0d5e52a48a064acab070a
June 6 2002	Alert.02.0606.gz d7ac0a20fa0f36ffa43d7978181f97ca	Scans.02.0606.gz 7d6029460f8ff0c4db3cd185543e40ab	Oos_Jun.6.2002.gz 9b9d6f780fa531ab6d165eba5158a837
June 7 2002	Alert.02.0607.gz d751af20ccb9b63ffbdfa426f8fa8fb6	Scans.02.0607.gz 878d3a183dfe1be1a241263bb491464c	Oos_Jun.7.2002.gz c456c36ddbe4fd953c23a47d5e6cfb75
June 8 2002	Alert.02.0608.gz c3bc308fa6c642c369dc549bd1c3fae4	Scans.02.0608.gz 9f01aba6909b4ee3c7790ccd8d114d4c	Oos_Jun.8.2002.gz 5056237b8d17479129dc65cbe92d7556
June 9 2002	Alert.02.0609.gz b65b2a84d011b5d30fe4b81fa1142a6f	Scans.02.0609.gz 9af735140419a2a22c695caa67add3cd	Null Value

3. Statement of Limitations

As it is not feasible to analyze all 229,000 individual alerts and 3,000,000 scans, our analysis focused on only the critical and significant alerts/scans/oos logs. Therefore our detailed analysis will primarily focus on significant exposures and vulnerabilities for the University network infrastructure.

4. Analysis

4.1 Table 1: Alert Summary: List Of Detects By Frequency

The following table shows a total list of unique detects as reported from the 5 days of alert logs we are analyzing. We have discovered a total of 75 alerts. We are not going to review every single alert but review the more interesting alerts that stand out. We know that a default Snort rule set contains many false positives and if not tuned specifically for a network environment can produce excess alerts that are unnecessary to review as they are part of normal network patterns.

No.	Signature (click for sig info)	# Alerts	# Sources	# Dests	% Total
1	SMB Name Wildcard	57794	182	281	25.2
2	spp_http_decode: IIS Unicode attack detected	51461	104	540	22.4
3	SNMP public access	30813	21	137	13.4
4	MISC Large UDP Packet	27711	12	9	12.1
5	ICMP Echo Request L3retriever Ping	26989	99	11	11.8
6	connect to 515 from inside	6848	32	3	3.0
7	INFO MSN IM Chat data	6064	94	96	2.6
8	WEB-MISC Attempt to execute cmd	3984	24	35	1.7
9	ICMP Echo Request Nmap or HPING2	3664	61	4	1.6
10	AFS - Off-campus activity	2609	55	19	1.1
11	High port 65535 udp - possible Red Worm - traffic	2355	104	129	1.0
12	Watchlist 000220 IL-ISDNNET-990517	1783	21	8	0.8
13	spp_http_decode: CGI Null Byte attack detected	1380	11	26	0.6
14	ICMP Router Selection	1007	109	1	0.4
15	ICMP Fragment Reassembly Time Exceeded	752	28	45	0.3
16	Null scan!	530	16	6	0.2
17	FTP DoS ftpd globbing	397	8	5	0.2

18 INFO Inbound GNUTella Connect request	337	293	4	0.1
19 ICMP Echo Request Windows	305	27	23	0.1
20 MISC Source Port 20 to <1024	297	1	297	0.1
21 WEB-IIS view source via translate header	285	8	1	0.1
22 ICMP Echo Request BSDtype	272	4	6	0.1
23 INFO Outbound GNUTella Connect request	245	4	159	0.1
24 WEB-MISC 403 Forbidden	178	7	11	0.1
25 SCAN Proxy attempt	152	10	19	0.1
26 SUNRPC highport access!	145	2	1	0.1
27 WEB-IIS Unauthorized IP Access Attempt	138	6	6	0.1
28 ICMP Destination Unreachable (Communication Administratively Prohibited)	136	3	3	0.1
29 IDS552/web-iis_IIS ISAPI Overflow ida nosize [arachNIDS]	133	128	30	0.1
30 INFO FTP anonymous FTP	112	4	19	0.0
31 NMAP TCP ping!	73	6	4	0.0
32 WEB-FRONTPAGE _vti_rpc access	56	19	2	0.0
33 WEB-IIS _vti_inf access	55	21	2	0.0
34 INFO Possible IRC Access	39	6	8	0.0
35 High port 65535 tcp - possible Red Worm - traffic	39	5	4	0.0
36 ICMP traceroute	39	13	4	0.0
37 WEB-MISC compaq nsight directory traversal	20	7	7	0.0
38 EXPLOIT x86 NOOP	18	9	9	0.0
39 UDP SRC and DST outside network	18	4	2	0.0
40 SCAN Synscan Portscan ID 19104	16	16	5	0.0
41 WEB-IIS 5 .printer isapi	16	2	3	0.0
42 INFO Inbound GNUTella Connect accept	16	1	3	0.0
43 INFO - Possible Squid Scan	15	6	6	0.0
44 Watchlist 000222 NET-NCFC	15	3	2	0.0
45 RFB - Possible WinVNC - 010708-1	13	6	5	0.0
46 MISC traceroute	12	4	3	0.0
47 EXPLOIT NTPDX buffer overflow	12	6	5	0.0

48	Port 55850 tcp - Possible myserver activity - ref. 010313-1	10	2	2	0.0
49	EXPLOIT x86 setuid 0	10	9	5	0.0
50	Attempted Sun RPC high port access	9	7	8	0.0
51	EXPLOIT x86 setgid 0	7	6	4	0.0
52	IDS553/web-iis_IIS ISAPI Overflow idq [arachNIDS]	7	1	6	0.0
53	Possible trojan server activity	7	3	3	0.0
54	TFTP - External UDP connection to internal tftp server	6	5	4	0.0
55	WEB-MISC http directory traversal	6	2	2	0.0
56	SCAN FIN	6	1	1	0.0
57	Queso fingerprint	5	3	4	0.0
58	EXPLOIT x86 stealth noop	4	1	3	0.0
59	ICMP Echo Request CyberKit 2.2 Windows	4	1	2	0.0
60	Back Orifice	4	4	4	0.0
61	Virus - Possible scr Worm	4	1	1	0.0
62	x86 NOOP - unicode BUFFER OVERFLOW ATTACK	3	3	3	0.0
63	WEB-CGI formmail access	3	3	1	0.0
64	WEB-IIS File permission canonicalization	2	1	1	0.0
65	ICMP Echo Request Sun Solaris	2	1	1	0.0
66	WEB-CGI scriptalias access	1	1	1	0.0
67	SMB CD..	1	1	1	0.0
68	IDS50/trojan_trojan-active-subseven [arachNIDS]	1	1	1	0.0
69	SYN-FIN scan!	1	1	1	0.0
70	Incomplete Packet Fragments Discarded	1	1	1	0.0
71	External RPC call	1	1	1	0.0
72	Probable NMAP fingerprint attempt	1	1	1	0.0
73	Virus - Possible pif Worm	1	1	1	0.0
74	WEB-CGI redirect access	1	1	1	0.0
75	WEB-IIS File permission canonicalization(Chinese charset)	1	1	1	0.0

4.2 Top 10 Alerts And Analysis Based On Frequency

No	*Name of Detect*	*Analysis of Detect*

1	**SMB Name Wildcard**	**Description:** SMB Name Wildcard are typically attempts to connect to a remote netbios name service. Windows machines and samba clients use SMB for netbios name to IP enumeration. SMB Name Service communicates on reflexive UDP port 137. The only concern with SMB Name Wildcard Traffic is if it originates from external networks. SMB is a very insecure protocol that does not provide "out of the box" authentication of its services . External SMB traffic is a sign of either attackers trying to enumerate windows hosts on your network (i.e using nbtstat) or unprotected windows hosts outside your network broadcasting requests. AS well numerous CVE entries have been posted: http://icat.nist.gov/icat.cfm?cvename=CVE-1999-0366 http://icat.nist.gov/icat.cfm?cvename=CAN-1999-0518 http://icat.nist.gov/icat.cfm?cvename=CAN-1999-0621 **Snort Rule that triggered alert:** alert UDP $EXTERNAL any -> $INTERNAL 137 (msg: "IDS177/netbios_netbios-name-query"; content: "CKAAAAAAAAAAAAAAAAAAAAAAAAAAAA|00 00|"; classtype: info-attempt; reference: arachnids,177;) **Example:** 06/05-00:00:16.670793 [**] SMB Name Wildcard [**] 130.85.11.6:137 -> 130.85.152.251:137 06/05-00:00:45.482428 [**] SMB Name Wildcard [**] 130.85.11.6:137 -> 130.85.152.159:137 **Recommendation:** According to Snort FAQ 4.15: Allowing netbios traffic outside University network is insecure. Ensure that University border routers/firewalls block tcp/udp ports 137,138,139 and 445 to prevent name querying and potential file transfers. Tune Snort IDS variable! $HOME to reflect University subnets thereby reducing internal false positive traffic. In addition we highly recommend implementing the RestrictAnonymous registry key for Internet-connected hosts in standalone or non-trusted domain environments. For more information see the following web pages: Windows NT 4.0: http://support.microsoft.com/support/kb/articles/Q143/4/74.asp Windows 2000: http://support.microsoft.com/support/kb/articles/Q246/2/61.ASP

2	**spp_http_decode: IIS Unicode attack detected**	**Description:** This detect was triggered by the snort http decode processor looking for Unicode character string sets. Specifically Microsoft IIS servers at one time were vulnerable to Unicode attacks that allowed attackers the ability to remotely read documents and execute arbitrary commands using Unicode strings. Correlations on this event can be found on **http://www.sans.org/top20.htm** see W1 for details and **http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2000-0884** However Snort is vulnerable to false positives caused by the http-decode preprocessor. Section 4.17 of the **Snort Faq** states: "Your own internal users normal surfing can trigger these alerts in the preprocessor. Netscape in particular has been known to trigger them." We have correlated our findings that support our analysis with Thomas Sheppards practical assignment. In addition we also noted that a majority of the destination IP addresses to these scans had 211.X.X.X IP notation. Most of the 211.x.x.x IP addresses belong to Asian Pacific registry network. This could explain the high volume of detections as the http preprocessor was picking up on Unicode / Asian character sets. Correlations on this event can be found on **http://www.sans.org/top20.htm see W1 for details** **Snort processor that triggered alert:** preprocessor http_decode: 80 8080 -unicode –cginull **Example:** 06/06-08:48:54.032163 [**] spp_http_decode: IIS Unicode attack detected [**] 130.85.153.118:3470 -> 211.32.117.38:80 06/06-08:48:54.032163 [**] spp_http_decode: IIS Unicode attack detected [**] 130.85.153.118:3470 -> 211.32.117.38:80 **Recommendation:** Microsoft has released a warning MS00-078 and a patch from MS00-057. This should correct any chance of this detect affecting internal web servers as that is the real issue. In addition to ensuring that all your web servers are patched with security fixes in a timely manner. To remove false positives the Snort FAQ recommends that Instead of disabling them,try a BPF filter to ignore your outbound http traffic such as: snort -d -A fast -c snort.conf not (src net 130.85.X.X and dst port 80) or we can simply ignore outbound web traffic to any Asian pacific networks 211.X.X.X.

3	**SNMP public access**	**Description:** Simple Network Management Protocol is a client server management tool defined by rfc 2771. The protocol operates on tcp/udp 161 and 162 and is platform agnostic. SNMP access is controlled via community strings. The default SNMP read community string is public and the default write community string by default is private. Community strings are in essence passwords that enable one to interact with the device on different levels. The signature is triggered when an SNMP request to udp port 161 is flagged with the "public" word is passed in traffic to a client. (Note: Client because SNMP roles in client/server are switched, the SNMP agent is the server) This enables a requesting client to query networking statistics/info/configs on the host running the service. Advisories have been recently submitted for SNMP version 1. SNMP is also listed on the SANS top 10 http://sans.org/top10 see U7 for details. As well numerous SNMP vulnerabilities are also noted in CAN-1999-0517, CAN-1999-0516, CAN-1999-0254, CAN-1999-0186. Fortunately we did not detect any outside networks targeting **internal SNMP agents. However our** **Example:** 06/05-00:01:01.647128 [**] SNMP public access [**] 130.85.70.177:1106 -> 130.85.5.96:161 06/05-00:01:01.653452 [**] SNMP public access [**] 130.85.70.177:1106 -> 130.85.5.96:161 **Recommendation:** We did not note any outside network attempts to query any internal SNMP agents. The following recommendations should be applied to all SNMP managed devices: • Ensure that SNMP version 2c is implemented across the University network, as they are many potential exploits to SNMP v1. As well use the appropriate Snort SNMP v2c plug-in to integrate IDS functionality • Change default public community string to something stronger and more obscure as system config and stats could be confidential information depending on the hosted applications (i.e. University administration and General Ledger servers) • Block TCP/UDP 161 and 162 at border choke points to prevent outside networks from querying internal servers.

4	**MISC Large UDP Packet**	**Description:** This detect was triggered by UDP packet sizes over 4000 bytes. Large UDP packets are flagged, as they can either be a sign of DDOS or transition of secure encrypted data. University logs appear to have multiple potential explanations to the cause. The source IP's 202.210.163.67 (**media**1.digi-c.com), 140.142.17.184 (wexler4.uw**tv**.washington.edu), 140.142.8.73, (**media**-wm-3.cac.washington.edu), and 207.25.79.241 (**realchannel**.cnn.com) as an examples to the use of multimedia streaming. This is evident in the DNS name associate to the host. This could explain potential large misc udp packets crossing the network. Another malicious example of Misc Large UDP packet comes from an IP 167.216.132.219 where the source and destination ports are set to 0 signifying an illegal packet. This could be a true sign of a DOS attack. Other examples should be identified on a case-by-case basis as this rule is vague and does not have a high degree of reliability upon closer review. Sources on this signature can be found at: http://www.whitehats.com/info/IDS247 **Snort Rule that triggered alert:** alert udp $EXTERNAL_NET any -> $HOME_NET any (msg:"MISC Large UDP Packet"; dsize: >4000; reference:arachnids,247; classtype:bad-unknown; sid:521; rev:1;) **Example:** 06/08-15:37:28.695316 [**] MISC Large UDP Packet [**] 202.102.12.29:6622 -> 130.85.153.117:4415 06/08-15:37:28.796794 [**] MISC Large UDP Packet [**] 202.102.12.29:6622 -> 130.85.153.117:4415 **Recommendation:** Misc UDP packets should be observed in correlation to other signs and not in isolation. When large UDP packets are substantiated with known Trojan ports like "netspy" or using illegal port addresses like 0, attention to those effected hosts should be investigated. As well by having a well-defined policy that permits specific media streaming sites that are enforced by perimeter routers and firewalls will drastically reduce the number of false positives returned by the IDS.

5	**ICMP Echo Request L3retriever Ping**	**Description:** The following detect was primarily designed to identify the use of the L3retreiver scanner. A network security scanner tool that identifies vulnerabilities across a network. The trigger is a ICMP ping request Type 8 Code 0 with a payload of ”AB…WABC…GHI”. As this represents 11% of all alerts, we must conclude that this is a false positive and identify other triggers. Arachnids supports this theory as a possible known false positive. **http://www.whitehats.com/info/IDS311** It is also known that Win2K controllers talk to workstations with the same pattern in the ICMP payload. We also can corroborate this assumption with the high alerting of SMB Name Wildcard associated with a windows network actively supports this theory. **Snort Rule that triggered alert:** alert ICMP $EXTERNAL any -> $INTERNAL any (msg: "IDS311/scan_ping-scanner-L3retriever"; itype: 8; icode: 0; content: "ABCDEFGHIJKLMNOPQRSTUVWABCDEFGHI"; depth: 32; classtype: info-attempt; reference: arachnids,311;) **Example:** 06/05-00:00:16.670344 [**] ICMP Echo Request L3retriever Ping [**] 130.85.152.251 -> 130.85.11.6 06/05-00:00:45.480550 [**] ICMP Echo Request L3retriever Ping [**] 130.85.152.159 -> 130.85.11.6 **Recommendation:** This detect should be disabled or at least modified. All traffic analyzed seemed to indicate normal non-malicious patterns. No external networks were detected as all detects were internal. The snort rule should be modified to ignore internally defined hosts by setting the “var HOME_NET” in snort.conf to read “var HOME_NET 130.85.0/16” In addition the snort message should be revised to read L3retriever /Win2k Controllers ICMP Echo Request) as to detect outside networks trying to communicate with internal Windows 2000 Domain Controllers.

6	**connect to 515 from inside**	**Description:** The following detect was triggered by attempt to access an Unix print spooler. Many advisories have been posted recently pertaining to un-patched LPRng software. See links for details. http://www.cert.org/advisories/CA-2000-22.htmlhttp://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2000-0917 http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2000-0615. The exploit of to a compromised server would allow an unauthorized user to execute arbitrary code on the server. The alerts account 3% of all logs in our set. Closer examination of the source and destination detects all happen to be internally directed traffic. We can note that in the past 5 days 32 unique sources were accessing 3 servers to port 515 the print spooler port (130.85.150.198, 130.85.153.191, and 130.85.5.35). I do not believe this is attributed to an un-patched Unix print server, as we would be expected to see supporting recon scans to first verify a server is running the printer port. As we cannot see evidence from recon scanning we can conclude with some certainty that these servers are legitimate print servers. **Example:** 06/05-13:17:48.705770 [**] connect to 515 from inside [**] 130.85.153.137:1269 -> 130.85.150.198:515 06/05-13:17:48.705838 [**] connect to 515 from inside [**] 130.85.153.137:1269 -> 130.85.150.198:515 **Recommendation:** Ensure that all print servers are running the latest security patches as a best practice recommendation. The snort signature should also be modified to only identify the source IP ranges that have legitimate access to communicate with the 3 target print serves. We can identify them as source IP's with the 130.85.152.0/24 and 130.85.153.0/24 range. All other traffic can be identified as suspicious and should be investigated.

7	**INFO MSN IM Chat data**	**Description:** This detect is triggered via clients using MSN Microsoft Network's Instant Messaging tool. Instant Messenger works with Microsoft Instant Messenger servers typically hosted by Microsoft. We can confirm that they are indeed MSN IM servers as the subnets we detected outside of the Universities internal networks are 64.4.12.0/24 and 64.4.13.0/24. These networks are registered to Microsoft with an "msgr.hotmail.com" domain. IM is a popular tool that is known to have many negative effects and exposures to internal organizational networks. MSN IM is also subject to a CERT advisories on an existing buffer overflow conditions in IM's chat active-X control. Due to lack of buffer checking it is possible to run arbitrary code within the user's privilege. More detail listed at: **http://securityresponse.symantec.com/avcenter/security/Content/1943.html** **http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2002-0155** **http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2002-0228** **http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2002-0377** **http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2002-0472** Other issues that are prevalent with Instant messaging is that all messages sent back and forth across the Universities network is in plaintext. Researchers and professionals working at the university might be ignorant to the inherent risks to using this common tool. IM is vulnerable to ease dropping and network monitoring. IM MSN is also capable to work through a firewall and capable to send files through existing open ports on firewalls and routers. In addition growing popularity to IM and related Peer to Peer applications can compete for network bandwidth otherwise better suited for education and research. **Snort Rule that triggered alert:** alert tcp $HOME_NET any -> $EXTERNAL_NET 1863 (msg:"INFO MSN chat access";flags: A+; content:"text/plain"; depth:100; classtype:misc-activity; sid:540; rev:3;) **Example:** 06/05-13:05:33.102143 [**] INFO MSN IM Chat data [**] 64.4.12.153:1863 -> 130.85.153.110:3759 06/05-13:06:36.103029 [**] INFO MSN IM Chat data [**] 64.4.12.153:1863 -> 130.85.153.110:3759 **Recommendation:** Ensure that all implementations of MSN IM are updated with the latest version of MSN Messenger. We also recommend that acceptable use policies be established to regulate ethical use of university computing resources if otherwise not implemented. Although technical solutions to control and secure IM are available (i.e. www.groove.net, www.parlano.com) they are still new and costly. If IM is not an acceptable tool, we recommend that you block known Microsoft IM Server IP addresses at external routers and

8	**WEB-MISC Attempt to execute cmd**	**Description:** The following detect strongly correlates to the "spp_http_decode" detect for source IP addresses as they seem to occur at the exact same time. What is apparent is that they both are attempting to exploit vulnerable Microsoft IIS servers. All source IP addresses are from external networks. The top 3 IP addresses (217.82.174.95, 141.76.1.121, 141.76.1.122) account for 75% of all source scans for these 2 alerts. David Stewart's practical suggests that this could be a variant of Code Red 2. David states the Code Red worm looks for systems running IIS that have not patched the unchecked buffer vulnerability in idq.dll or removed the ISAPI script mappings. The worm exploits the vulnerability to inject itself into a system. No evidence that any source IP with the Universities Internal addresses was found exhibiting signs of a compromised host. Therefore we can only report on attempts and not actually infected University IIS web servers **Example:** 06/09-03:41:05.591049 [**] spp_http_decode: IIS Unicode attack detected [**] 217.82.174.95:3413 -> 130.85.5.14:80 06/09-03:41:05.591049 [**] WEB-MISC Attempt to execute cmd [**] 217.82.174.95:3413 -> 130.85.5.14:80 **06/09-03:41:05.604603** [**] spp_http_decode: IIS Unicode attack detected [**] 217.82.174.95:3414 -> 130.85.5.14:80 **06/09-03:41:05.604603** [**] WEB-MISC Attempt to execute cmd [**] 217.82.174.95:3414 -> 130.85.5.14:80 Note: The following detect triggers also triggered spp_http_decode: IIS Unicode attack detected. **Recommendation:** It is good practice to ensure that all Microsoft IIS Web Servers have been patched with the latest IIS server security patches in a timely manner Ensure that the University creates and maintains documented hardening standards for the builds and rebuilds of all Microsoft web servers to ensure that they all have the same baseline level of security. Periodicallyrun the Microsoft baseline security analyzer to assess your results.http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/tools/Tools/MBSAhome.asp . Lastly ensure that Antivirus software is installed on all Windows machines and they are running the latest definition files from the vendor.

9	**ICMP Echo Request Nmap or HPING2**	**Description:** The following detect was triggered as a result of command line network tools used for testing, profiling, and mapping remote networks using ICMP Type 8 Code 0 requests. Closer examination of the logs show that 61 internal sources are communicating with 4 destination hosts. The result is not 61 hosts using Nmap or Hping2 but a possible false positive on Windows clients communicating with Windows Active Domain Controllers and one external host. The following lists are resolved IP addresses of the Universities Windows Servers. Note that the name of the host is also indicative or its function. (dc=domain controller, ad=active directory) 130.85.11.7 (dc2.ad.UMBC.EDU) 130.85.11.6 (dc1.ad.umbc.edu) 130.85.1.3 (UMBC3.UMBC.EDU) **209.53.113.23 (m23.absolute.com) -could be legitimate Nmap/Hping2** **Example:** 06/05-00:02:26.499381 [**] ICMP Echo Request L3retriever Ping [**] 130.85.152.176 -> 130.85.11.7 06/05-00:05:57.409713 [**] ICMP Echo Request L3retriever Ping [**] 130.85.152.176 -> 130.85.11.7 **Recommendation:** This detect similar to the recommendations advised for detect no. 5 (ICMP Echo Request L3retriever Ping). The snort rule should be modified to ignore internally defined hosts by setting the "var HOME_NET" in snort.conf to read "var HOME_NET 130.85.0/16" thereby eliminating unnecessary network noise..

10	**AFS - Off-campus activity**	**Description**: The following detect is an alert triggered by AFS. AFS stands for the Andrew File System and is a distributed file system that offers file-sharing capability. AFS operates by having the server listen on port 7000 and clients connect from port 7001. RFC 1340 contains all AFS defined ports: http://www.faqs.org/rfcs/rfc1340.html. The code can be found on openafs.org. OpenAFS, an open source freeware version of the code is vulnerable to a buffer overflow that bug in the RPC library used by OpenAFS that could be exploited to crash certain OpenAFS servers. It is also possible to obtain unauthorized root access to a host running one of these processes (OPENAFS-SA-2002-001 - xdr_array integer overflow) See http://www.openafs.org/security/ for more details. As in any distributed file sharing applications, it is critical to ensure data integrity, confidentiality and security of University information. Universities are but special exceptions in that sharing of files or file systems is something that security controls should not prevent, but detect and correct without limiting access to University resources. We can correlate this by observing the off campus AFS activity submitted. Most offsite AFS servers are mostly other Universities sharing information. **Example:** 06/06-16:49:56.649417 [**] AFS - Off-campus activity [**] 64.15.254.25:7000 -> 130.85.152.141:7001 06/06-16:49:57.945824 [**] AFS - Off-campus activity [**] 64.15.254.25:7000 -> 130.85.152.141:7001 **Recommendation:** We recommend that all University computers should have Antivirus software installed to check for inbound and outbound viruses. Ensure if any AFS servers are hosted that they are utilizing the latest version to date for OpenAFS. Documented Acceptable Use policies should be communicated to ensure that AFS is used according to University policies and not used for hosting unauthorized materials, software, and music.

4.3 Top Ten Talkers for "Alerts" Logs By Source IP Internal Hosts

Rank	Total # Alerts	Source IP	# Signatures triggered	Destinations involved
#1	13218 alerts	130.85.11.6	1 signatures	(48 destination IPs)
#2	12841 alerts	130.85.70.177	2 signatures	(28 destination IPs)

#3	12355 alerts	130.85.11.7	1 signatures	(51 destination IPs)
#4	5952 alerts	130.85.88.154	3 signatures	(24 destination IPs)
#5	4094 alerts	130.85.150.198	2 signatures	(105 destination IPs)
#6	3462 alerts	130.85.153.120	2 signatures	(28 destination IPs)
#7	3372 alerts	130.85.5.89	1 signatures	(170 destination IPs)
#8	2639 alerts	130.85.88.201	1 signatures	(44 destination IPs)
#9	2367 alerts	130.85.153.114	2 signatures	(13 destination IPs)
#10	2364 alerts	130.85.150.245	1 signatures	130.85.152.109

Criteria: Based on number of source IP occurrences. Top 10 alerts are useful to determine areas of high volume generating normal and abnormal traffic on the network. For example we can deduce that internal source IP addresses with few signatures, many destinations can be an easy method of detecting critical university services. Understanding the relationships between the top 10 source IP alerts can be beneficial in learning where the bulk of detects is originating from.

4.4 Top Ten Talkers for "Alerts" Logs By Destination IP Internal Hosts

Rank	Total # Alerts	Destination IP	# Signatures triggered	Originating sources
#1	28131 alerts	130.85.11.6	4 signatures	(48 source IPs)
#2	26531 alerts	130.85.11.7	3 signatures	(52 source IPs)
#3	12517 alerts	130.85.152.20	4 signatures	(8 source IPs)
#4	11242 alerts	130.85.153.117	3 signatures	130.34.64.6, 202.102.12.29
#5	9192 alerts	130.85.150.195	6 signatures	(32 source IPs)
#6	6845 alerts	130.85.150.198	3 signatures	(32 source IPs)
#7	4450 alerts	130.85.5.96	14 signatures	(57 source IPs)
#8	3234 alerts	130.85.5.97	5 signatures	(13 source IPs)
#9	2644 alerts	130.85.152.109	1 signatures	(4 source IPs)
#10	2558 alerts	130.85.5.127	3 signatures	(3 source IPs)

Criteria: Based on number of destination IP occurrences. Top 10 alerts for destination IP addresses are useful in identifying the potential key/significant hosts on the internal University network. This will include file and print services, domain controllers and hosts that are running multiple services and are potentially critical services for the university.

4.5 Top Ten Talkers for "Scans" Logs By Source Host/Port

Rank	Source Host	Number of Scan Records
#1	130.85.88.162	876063
#2	130.85.5.89	601890

#3	130.85.60.43	349700
#4	130.85.11.8	184251
#5	130.85.153.191	178787
#6	130.85.153.190	32742
#7	130.85.6.45	29275
#8	130.85.6.49	29181
#9	130.85.6.51	24127
#10	130.85.6.50	23636

Rank	Source Port	Name	Number of Scan Records
#1	1214	Kazaa	1063504
#2	1111	Unknown	598968
#3	123	Network Time Protocol	315145
#4	1347	bbn-mmc Multimedia Conferencing	184394
#5	7000	AFS3-fileserver	98773
#6	7001	AFS3-callback	93160
#7	137	NetBios Name Server (reflexive port)	51355
#8	0	Illegal port used to fingerprint OS	42767
#9	6970	RTP (Real Player Real Audio)	27116
#10	88	Kerberos, WWW	15184

Criteria: Based on the number of most active hosts scanning, we can potentially identify significant activity between hosts 130.85.88.162 and 130.85.5.89. As well we can identify multiple services that correlate to alerts analyzed above. For example we see Peer to Peer, AFS, SMB Name Wildcard and Realplayer (Misc Large UDP Packets) are examples of a few.

4.6 Top Ten Talkers for "Scans" Logs By Destination Host/Port

Rank	Destination Host	Number of Scan Records
#1	130.85.1.3	48145
#2	130.85.11.6	26184

#3	130.85.11.7	25754
#4	130.85.1.4	24674
#5	130.85.6.45	19779
#6	130.85.60.43	16281
#7	130.85.88.245	15655
#8	130.85.5.55	15351
#9	130.85.5.50	15027
#10	130.85.152.15	14672

Rank	Destination Port	Name	Number of Scan Records
#1	1214	Kazaa	1057708
#2	161	SNMP	600350
#3	1346	Alta-Analytics License Manager	184482
#4	80	Web	170588
#5	7001	AFS	98788
#6	53	DNS	74474
#7	7000	AFS	71793
#8	137	Netbios Name Server	51262
#9	0	Reserved	35520
#10	6970	RTP (Real Player Real Audio)	35089

Criteria: Based on the number of most active hosts scanning, we can potentially identify significant activity to internal hosts 130.85.1.3 and 130.85.11.6. As well we can identify multiple services that correlate to alerts analyzed above. For example we see Peer to Peer, AFS, SMB Name Wildcard and Realplayer (Misc Large UDP Packets) are examples of a few. We can also correlate the large SNMP alerts with high scan records to port 161.

4.7 All Talkers for Oos Logs By Source and Destination IP's

Rank	Source IP	Destination IP	# of Occurrences
#1	24.226.42.77	130.85.153.150	9 times
#2	195.101.94.208	130.85.5.95	3 times
#3	216.150.166.88	130.85.153.189	1 time
#4	212.111.92.2	130.85.153.189	1 time
#5	24.65.17.116	130.85.88.178	1 time

Criteria: Dues to the extremely small size of the logs, we are able to represent all entries. As Oos detects are triggered by unconventional packet crafting. The university should still investigate the #1 entry as repeated attempts can signify malicious behavior. Interesting to note are the repeated and reused patterns of source and destination port assignments.

=+

```
06/04-18:59:58.963952 24.226.42.77:2331 -> 130.85.153.150:6346
TCP TTL:109 TOS:0x0 ID:28026  DF
21**R**U Seq: 0x83   Ack: 0xB9160017   Win: 0x5010
TCP Options => EOL EOL EOL EOL EOL EOL SackOK SackOK SackOK EOL Opt 20 Opt 20 Opt
20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20
Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt 20 Opt
20
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
06/04-19:00:13.470197 24.226.42.77:0 -> 130.85.153.150:2331
TCP TTL:109 TOS:0x0 ID:56955  DF
2*SF**AU Seq: 0x18CA0083   Ack: 0xCA320018   Win: 0x5010
18 CA 00 83 CA 32 00 18 18 73 50 10 22 21 29 73  .....2...sP."!)s
00 00 00 00 00 00                                ......
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
06/04-19:00:23.017204 24.226.42.77:0 -> 130.85.153.150:2331
TCP TTL:109 TOS:0x0 ID:55420  DF
21SFR*AU Seq: 0x18CA0083   Ack: 0xCA320018   Win: 0x5010
32 F7 50 10 20 63 10 AD 00 00 00 00 00 00        2.P. c........
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
06/04-19:00:24.229749 24.226.42.77:2331 -> 130.85.153.150:6346
TCP TTL:109 TOS:0x0 ID:63868  DF
21SFR*AU Seq: 0x83   Ack: 0xCA320018   Win: 0x5010
35 F7 50 10 1D 63 10 AD 00 00 00 00 00 00        5.P..c........
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

This could indeed be a sign of attempting to scan a Trojan machine or attempting a DoS on the target host. We know that this is a crafted packet as the packets contain unconventional characteristics like non-incremental ack numbers/source port 0/ multiple TCP options set and almost all TCP flags set.

4.8 Five External Hosts Analyzed

No.	Source IP/DNS	Description and Registration Info

1	**167.216.132.219 wmvip-s20000005201.fplive.net**	The following host was chosen as this host was the source of malicious traffic sending large UDP traffic to the reflexive broadcast address of port 0 (Port 0 signifies an illegal packet and a potential DOS). Dshield also can corroborate our decision to place this as a host to be investigated as it is been reported 54 times from 20 targets. **http://isc.incidents.org/source_report.html?subnet=167.216.132.219** We can also correlate this same combination of signatures with GIAC practical Angela Orebaugh who had the same signature. **Arin: 167.216.132.219** Digital Island, Inc. (NETBLK-MIC-DIGISLE-A) 45 Fremont St, Suite 1200 San Francisco, CA 94105 US Netname: MIC-DIGISLE-A Netblock: 167.216.128.0 - 167.216.143.255 Maintainer: DIIS Coordinator: Digital Island, Inc. 45 Fremont Street (NR-ORG-ARIN) netreg@digisle.net 415.738.4100 **NSI: fplive.net** Sandpiper Networks (FPLIVE-DOM) 225 West Hillcrest Dr., Suite 150 Thousand Oaks, CA 91360 US Domain Name: FPLIVE.NET Administrative Contact: Streaming Support (SS7719-ORG) alangley@EXODUS.NET Exodus 225 West Hillcrest Dr., Suite 150 Thousand Oaks, CA 91360 US 805-370-2100 Fax- 805-370-2101 Technical Contact: Support (GJXIQFNILO) support@DIGISLE.COM Digital Island, Inc. 225 West Hillcrest Drive, #150 Thousand Oaks , CA 91360 US 877-885-5550 Fax- 805-370-2181

2	**130.34.64.6 mat-hub2.material.tohoku.ac.jp**	The following host was identified as it was detected systematically scanning 297 internal University hosts all from port 20 on the source IP and port 360 on the destination IP. The scan surface on June 9 2002 from 12:30pm -8:00pm and automatically scanned approximately every 1.6 minutes. Most likely a automated scanner looking for installed Trojans listening on this irregularly low destination port. ARIN: 130.34.64.6 Tohoku University (NET-TAINS) Katahira, Aoba-Ku980-77 JP Netname: TAINS Netblock: 130.34.0.0 - 130.34.255.255 Coordinator: Sone, Hideaki (HS206-ARIN) tains@tains.tohoku.ac.jp +81-22-217-6091 (FAX) +81-22-217-6098 JPNIC WHOIS: tohoku.ac.jp Domain Information: a. [Domain Name] TOHOKU.AC.JP g. [Organization] Tohoku University l. [Organization Type] National University m. [Administrative Contact] HA683JP n. [Technical Contact] HS024JP n. [Technical Contact] MC002JP p. [Name Server] ns1.tohoku.ac.jp p. [Name Server] ns2.tohoku.ac.jp y. [Reply Mail] tains@tains.tohoku.ac.jp [State] Connected (2003/03/31) [Registered Date] [Connected Date] [Last Update] 2002/06/12 14:58:35 (JST) kawa@topic.ad.jp

3	**166.102.16.18 h166-102-016-018.ip.alltel.net**	The following host was identified as it was executing "WEB-MISC Attempt to execute cmd" on 23 unique hosts in 40 minutes on June 9 2002. This detect is significant as it was a producing attack patterns of a dos storm worm seeking out vulnerable Microsoft IIS severs. www.incidents.org/react/dosstormworm.php As well this source IP can be linked to the Internet Storm Center http://isc.incidents.org/source_report.html?subnet=166.102.016.018. **ARIN: 166.102.16.18** ALLTEL Corporation (NET-ALLTEL) 1 Allied Dr Little Rock, AR 72202 US Netname: ALLTEL Netblock: 166.102.0.0 - 166.102.255.255 Maintainer: ALLT Coordinator: Services Hostmaster, Alltel Internet (AIS2-ARIN) hostmaster@alltel.net 501-905-4274 **NSI: Alltel.net** Alltel Information Services (ALLTEL2-DOM) 4001 Rodney Parham Rd Little Rock, AR 72212 Domain Name: ALLTEL.NET Administrative Contact, Technical Contact: Support, Technical (ST109-ORG) hostmaster@ALLTEL.NET ALLTEL Internet Prod Mgmt Department 1 Allied Drive Building V Floor 9 Little Rock, AR 72202 US 501-905-8000 Fax- - 501-905-6777 Fax- - - 501-905-7901

4	**4.64.196.126 (snjpca1-ar4-4-64-196-126.snjpca1.dsl-verizon.net)**	The following host should be identified as it is an external DSL user that has been detected attempting to scan for vulnerable University IIS web servers. On June 6 2002 for 90 minutes, the source attempted to exploit IIS 24 instances of spp_http_decode: IIS Unicode attack and 48 instances of WEB-MISC Attempt to execute cmd all on known IIS web servers. **ARIN: 4.64.196.126** GTE Intelligent Network Services (NETBLK-GTEINS-196-20) 5525 MacArthur Ste 320 Irving, TX 75038 US Netname: GTEINS-196-20 Netblock: 4.64.196.0 - 4.64.197.255 Coordinator: Hostmaster, Verizon Online (VOH1-ARIN) hostmaster@bizmailsrvcs.net 800-927-3000 Registrant: Verizon Trademark Services, LLC (VERIZON2-DOM) 600 Hidden Ridge Drive Irving, TX 75038 US **NSI: dsl-verizon.net** Domain Name: VERIZON.NET Administrative Contact: Andersen, Christian (CAH535) christian.andersen@VERIZON.COM Verizon Corporate Services Group Inc. 600 Hidden Ridge Drive Mailcode HQE03H01 Irving, TX 75038 US 972.718.7621 972.718.3946 Technical Contact: Hostmaster (HO9610-ORG) hostmaster@BIZMAILSRVCS.NET Verizon Online 5525 MacArthur Ste 320 Irving, TX 75038 US 800-927-3000

5	**63.121.84.239 (Does not resolve with DNS name)**	The following should be investigated as it was connecting to a known internal host that was detected running an active subseven Trojan. The fact that the detect is not a scan or a probe further warrants investigation as these types of detects are not spoofed IP addresses but originate from legitimate sources. ARIN: ISP Alliance (NETBLK-UU-63-121-80) 120 Milledgeville Highway Gordon, GA 31031 US Netname: UU-63-121-80 Netblock: 63.121.80.0 - 63.121.87.255 Coordinator: Abbott, Mike (MA517-ARIN) rgenovese@ispalliance.net 912-628-6000 x3203

4.9 Insight to Internal Hosts Compromised/Anomalous Activity

The following internal hosts are show probable signs of compromise and or demonstrate patterns of possible dangerous activity. 130.85.153.196, 130.85.152.181, 130.85.153.160, and 130.85.152.22 all share the widely common Back Orifice Server Port (Port 31337). Back Orifice is a widely known Trojan that allows an attacker to remotely control a server. The four internal University hosts that have triggered these detects are 130.85.6.49-52.

06/05-11:45:54.111406 [**] Back Orifice [**] 130.85.6.49:26465 -> 130.85.153.196:31337
06/06-09:43:16.615605 [**] Back Orifice [**] 130.85.6.50:26465 -> 130.85.152.181:31337
06/07-17:16:09.675473 [**] Back Orifice [**] 130.85.6.51:25193 -> 130.85.153.160:31337
06/07-14:53:38.994959 [**] Back Orifice [**] 130.85.6.52:14638 -> 130.85.152.22:31337

Note: Although BO is sometimes identified as a legitimate admin tool, we still believe these hosts should be investigated by University network administrators to verify that they are not compromised.

Other hosts that were seen to have exhibited signs of compromise are 130.85.151.107 and 130.85.150.133. These are referenced below and represented as part of our link graph model. Host 130.85.151.107 was selected as potentially compromised as it exhibited multiple malicious signature patterns from numerous outside hosts and attempting to connect to nonstandard ports that are not associated to standard TCP/IP services as referenced in RFC 1340. In addition the times for which these detects were triggered

appear to be during times at which are considered off-peak university hours. A known time where attackers are known to be least detected as network staff and support is low.

06/05-00:25:44.267954 [**] EXPLOIT x86 NOOP [**] 63.214.183.10:36040 -> 130.85.151.107:879
06/05-00:33:48.908520 [**] EXPLOIT x86 setgid 0 [**] 63.214.183.10:36046 -> 130.85.151.107:879
06/05-17:11:58.027753 [**] SCAN Proxy attempt [**] 216.120.51.60:1395 -> 130.85.151.107:1080
06/05-19:20:42.871259 [**] x86 NOOP - unicode BUFFER OVERFLOW ATTACK [**]
63.214.183.10:37115 -> 130.85.151.107:879
06/05-23:54:18.661789 [**] EXPLOIT x86 setgid 0 [**] 64.30.119.138:41957 -> 130.85.151.107:879
06/06-00:56:03.351550 [**] EXPLOIT x86 setgid 0 [**] 64.30.119.138:42172 -> 130.85.151.107:879
06/07-02:53:29.566477 [**] EXPLOIT x86 setgid 0 [**] 66.79.131.45:65302 -> 130.85.151.107:989
06/09-13:50:02.583658 [**] MISC Source Port 20 to <1024 [**] 130.34.64.6:20 -> 130.85.151.107:360

Host 130.85.150.133 also appears to be compromised as it was triggered by 30 distinct source IP addresses (all from external networks) with 11 different signature patterns. Some of the more interesting signature patterns are:

ICMP Destination Unreachable (Communication Administratively Prohibited) – a sign that the target IP has attempted to reach a host but a router with access control is preventing or blocking the request hence the router is sending back the following message using ICMP.
06/06-12:18:26.242111 [**] ICMP Destination Unreachable (Communication Administratively Prohibited) [**] 198.26.144.6 -> 130.85.150.133

Multiple Null Scans: - a sign of reconnaissance to the target host before an attack
06/05-09:20:17.946160 [**] Null scan! [**] 66.218.233.74:0 -> 130.85.150.133:1668
06/05-09:20:17.946160 [**] Null scan! [**] 66.218.233.74:0 -> 130.85.150.133:1668

And an IIS ISAPI Overflow: - attempt to exploit well known vulnerabilities in an IIS Microsoft web server
06/05-14:39:41.833283 [**] IDS552/web-iis_IIS ISAPI Overflow ida nosize [**] 212.2.214.54:3974 -> 130.85.150.133:80

4.10 Correlation of Other Practical

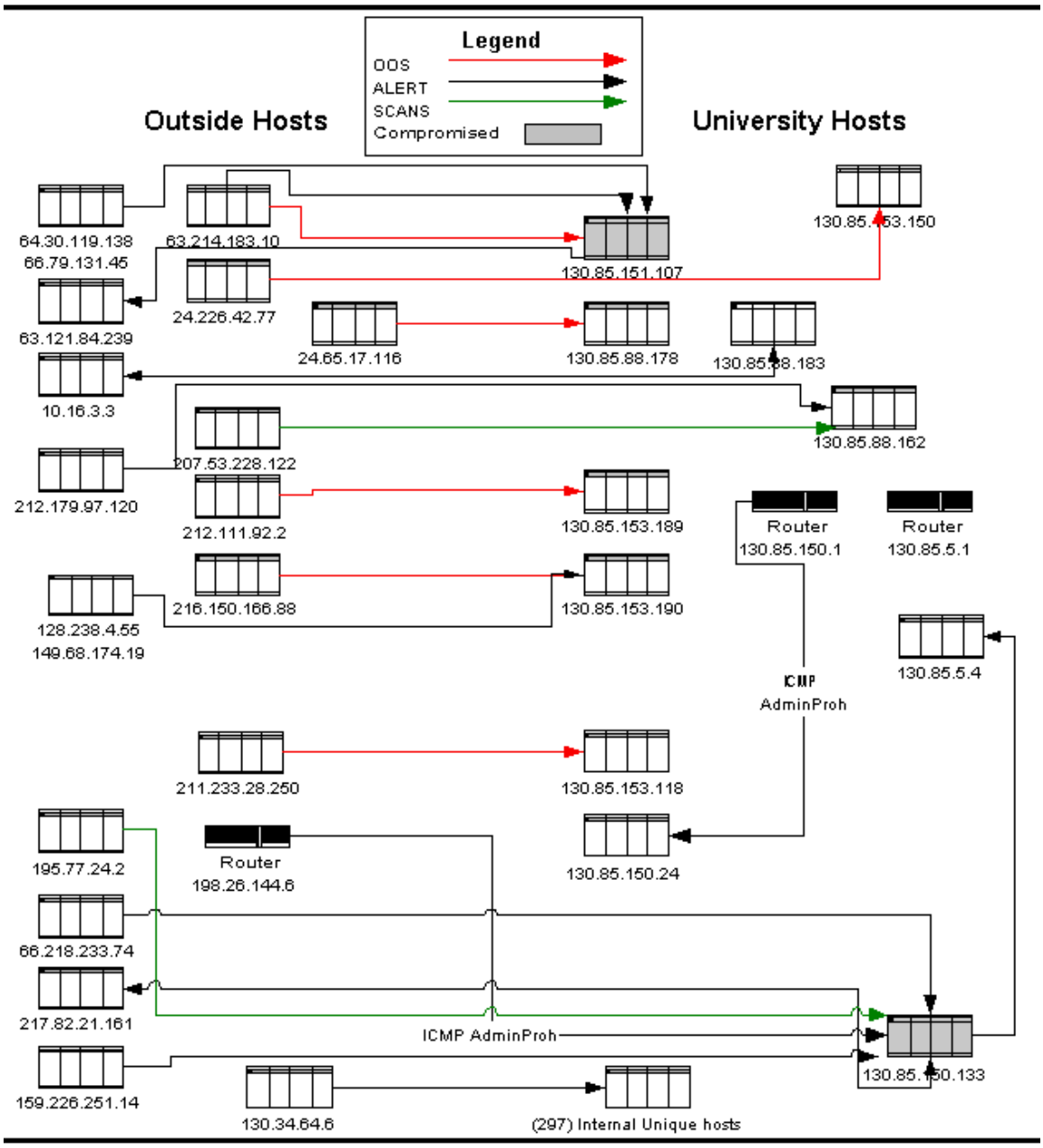
Referenced throughout the entire practical.

4.11 Link Graph

The following link graph below will attempt to address a relationship among malicious scans, alerts and out of spec logs from outside networks to the University managed network. The selected links were chosen on the following criteria: quantity of instances, known malicious signature with low false positive probability, and direction of traffic must either be going from or coming into the University network (no internal – internal detects reviewed). Also we incorporated the very few out-of-spec detects in the following link graph to highlight and correlations to the other log types. It was interesting to see that some of the OOS detects related to Alerts detect patterns as well. Please note that this is

not a representation of all malicious traffic flow but an example on how different representation of tabular data can highlight relationships that otherwise might be missed altogether using statistics and predetermined queries on known signature patterns.

From the graph we should note that there does exist relationships between OOS scans and Alerts by observing common destination hosts. We have highlighted 2 hosts that from the link graph appear to be compromised by the type of alert and direction of the traffic. We can also determine critical routers in the environment by observing ICMP admin prohibited replies and see that some hosts trigger detects in a bi-directional flow. This could indicate a combination of reconnaissance detects as well as successful compromises. See host 130.85.150.133 as an excellent example.



4.12 Abstract of Analysis Methodology

The analysis methodology used comprised of several well-documented tools; snortsnarf with install instructions provided by silicondefence.com. Snortsnarf utilizes perl and php to format and parse the large sums of data provided by the University. I recommend that you attempt this with as much ram in your wintel box allowed. I used around a gig of ram on my 800Mhz box and snortsnarf took over 3 hours to run through the logs. Scan logs were analyzed by importing the logs into MS Access and building simple queries to sort and view the data. The Out-Of-Spec files yielded very few entries (<20) and could easily be reviewed without any tools.

4.13 Bibliography

[Northcutt and Novak, 2001] Stephen Northcutt and Judy Novak. Network Intrusion Detection An Analyst's Handbook Second Edition. New Riders Publishing, Second Edition, Indianapolis, 2001.

[Cooper, Fearnow, Frederick and Northcutt, 2001] Stephen Northcutt, Mark Cooper, Matt Fearnow, and Karen Frederick. Intrusion Signatures and Analysis. New Riders Publishing, First Edition, Indianapolis, 2001.

[Mclure, Scrambray and Kurtz, 1999] Stuart Mclure, Joel Scambray, and George Kurtz. Hacking Exposed, Network Security Secrets & Solutions. Osbourne/McGraw Hill, Berkley, 1999.

[Cheswick and Bellovin, 1995] William R. Cheswick and Steven M. Bellovin. Firewalls and Internet Security: Repelling the Wily Hacker. Addison-Wesley Professional Computing Series, Massachusetts, 1995.

[Mandia and Prosise, 2001] Kevin Mandia and Chris Prosise. Incident Response, Investigating Computer Crime. Osbourne/McGraw Hill, Berkley, 2001.

[Berg, 2001] Al Berg. P2P, or Not P2P. Information Security Magazine. February 2001.

Shepherd, Thomas GIAC Practical URL:
http://www.giac.org/practical/Thomas_Shepherd_GCIA.doc GCIA ID 0442

Orebaugh, Angela GIAC Practical URL:
http://www.giac.org/practical/Angela_Orebaugh_GCIA.doc GCIA ID 0498

Stewart, David GIAC Practical URL:
http://www.giac.org/practical/david_stewart_gcia.doc GCIA ID 0468

Steele, Michael. Technical Support Silicone Defense SnortSnarf for Windows URL: http://www.siliconedefense.com/techsupport/winsnortsnarf-iis_1.8.7.htm

Black, Kevin. GIAC Practical URL: http://www.giac.org/practical/Kevin_Black_GCIA.doc GCIA ID 0273