SANS GCIA Practical Assignment

Charles Pham
September 2002
Version 3.3

Table of Contents

Assignment #1: Describe the State of Intrusion Detection

Deployment Planning of IDS in the Enterprise

Introduction

Over the years, Intrusion Detection System has become the buzzword in the security industry after it has been demonstrated that firewall was not the mean to end all security related attacks. In theory, the IDS provide a second layer defense in the event that the firewall is breached. However, in practice, IDS has several shortcomings that if incorrect managed would mean money down the drain. Assuming the enterprise already has a comprehensive Intrusion Detection System policy in place, the following provides information that will help in carrying out an enterprise wide IDS deployment.

IDS Technology

<u>Network-based IDS</u>

Network Intrusion Detection System monitor network traffic in the network segment that they are connected to and trigger alerts based on the following principles: pattern-matching, threshold violation, and anomaly detection.

Typical NIDS consists of two components: a sensor and a management console. The sensor is placed on the network segment to be monitored and report alerts back to the console. In most implementations, the sensor is placed behind a filtering device such as a firewall or router. However, there are special circumstances, such as field studies or as early warning system, when the sensor is deployed in front of a filtering device. Keep in mind that placing the sensor in front of a filtering device in an enterprise class network will require significantly more resource to operate. The console is placed in a secured network segment and is used to manage multiple sensors.

Advantages

Cost effective deployment in an enterprise
Operating system independence
Attack evidence preservation
Enabled proactive response

Disadvantages

False positive

Does not work on encrypted traffic
Higher false negative as traffic load increases
Scalability
Product centric security
Less effective in detecting trusted insider attack

Host-based IDS

Host Intrusion Detection System monitor system audit and events logs through system agent software that is install on the host and trigger alerts based on violation of specific set of rules.

Typical HIDS consists of two components: a software agent and a management console. The software agent is placed on the host to be monitored and report alerts back to the console. Deployment of HIDS agent in an enterprise network can prove to be a daunting task and as a result are often limited to only critical servers. Similar to the NIDS, the console is usually placed in a secured network segment and is used to manage multiple agents.

Advantages

Less false positive
No additional hardware investment
Works with encrypted traffic
Effective detection of trusted insider attack

Disadvantages

Vulnerable to direct attack
Cost of deployment in an enterprise
Reactive response only capability
Scalability
Product centric security

Hybrid IDS

Hybrid Intrusion Detection System combines the functionality of both NIDS and HIDS, in that it will monitor network traffics, system audit and event logs through agent software on the host. Alerts are raised by the combination or subsets of both NIDS and HIDS triggering mechanisms.

Typical Hybrid IDS consists of two components: a software agent and a management console. The software agent is placed on the host residing on the network to be monitored and report alerts back to the console. Similar to the HIDS in the enterprise, deployment of the software agent is often limited to critical servers. The console is usually placed in a secured network segment and is used to manage multiple agents.

Advantages

Less false positive
No additional hardware investment
Works with encrypted traffic
Enabled proactive response
Effective detection of trusted insider attack

Disadvantages

Vulnerable to direct attack
Cost of deployment in an enterprise
Scalability
Product centric security

Meta IDS

Meta Intrusion Detection System aggregates alerts from all the security devices, mine and correlate the raw data for attack information and present them in useful format. Alerts are raised by a combination of predefined or user defined rule sets.

Typical Meta IDS consists of a single component: a management system that is capable of accepting alerts and logs from a variety of devices. Information is collected from existing network capable devices. In order to keep up with the massive amount of data, the system can consists of multiple machines that function as collector, storage, transformation, warehouse, mining, query and analysis, and finally presentation. The whole system is usually placed in a secured network segment with input interfaces connected to the enterprise network.

Advantages

Less false positive
Works with encrypted traffic
Enabled proactive response
Attack evidence preservation
Operating system independence
Scalable
Process centric security
Detect unpublicized attack
Effective detection of trusted insider attack

Disadvantages

High up-front cost
Emerging technology
Required in-house development

Architecture

For the sake of simplicity, filtering devices are implied and are not shown between network segments within the enterprise.

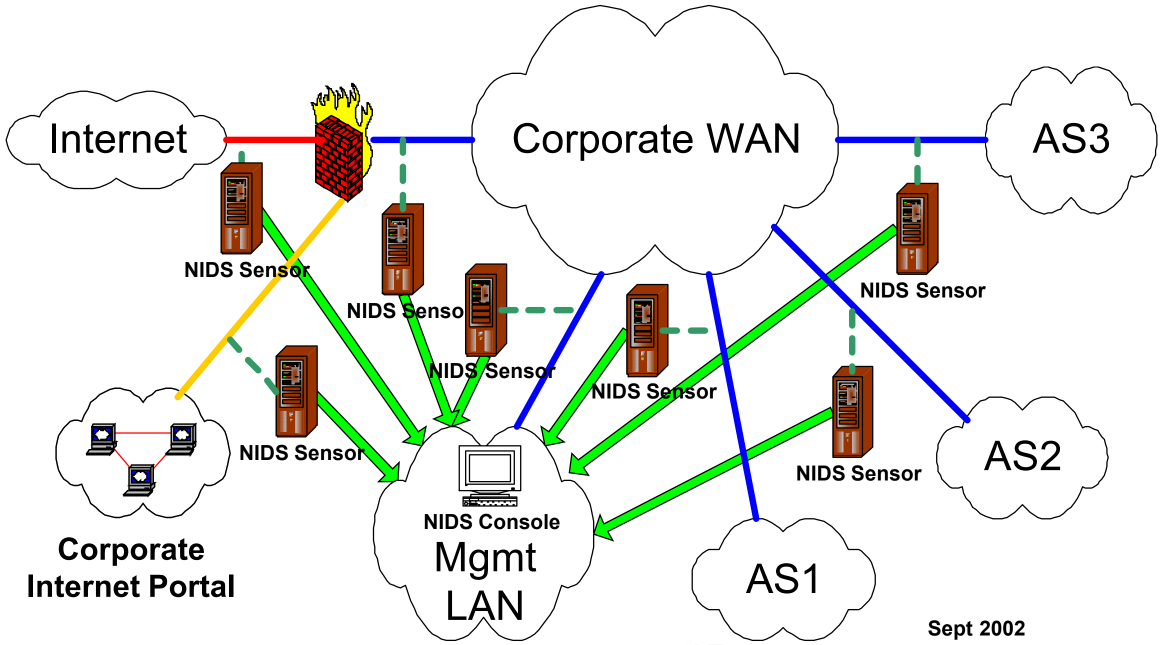
In addition, it should be noted that communication between sensor and management console is ideal if strong encryption and authentication are involved.

Network-based IDS

The diagram below illustrates the ideal type of architecture and the typical locations where NIDS sensors would be placed. Deployment of NIDS is most effective in network where system boundaries and topologies are well defined.

A common practice for deployment of network sensors is to make the monitoring network interface as stealthy as possible to avoid direct attack to the NIDS. This is illustrated below by the dotted green line.

Connectivity between the NIDS sensors and console needs to be well protected via a dedicated physical management network, VPN if the physical connection is shared with other network devices, or both. Connectivity requirements (e.g. ensuring the VPN will be able to pass through filtering devices residing between network segments, especially if Network Address Translation is in use) must be considered at the planning stage in order to avoid costly sidetrack during the deployment stage.

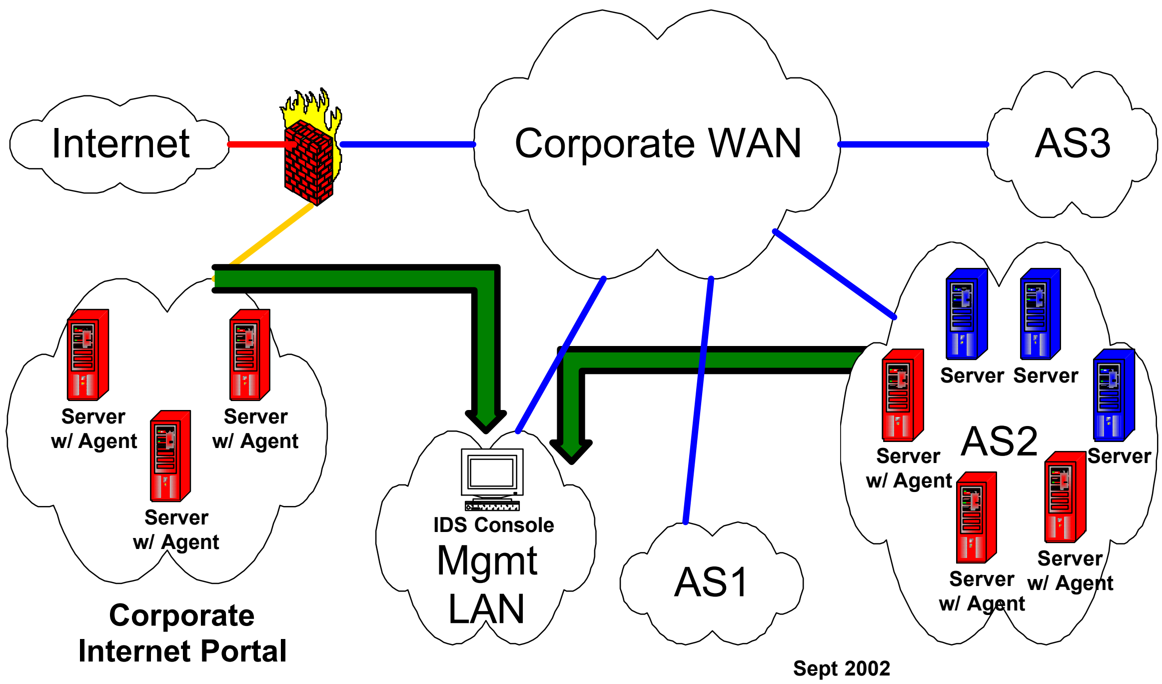


Host-based / Hybrid IDS

The diagram below illustrates the ideal type of architecture and the typical locations where HIDS sensors would be placed. The Hybrid IDS also shared the same type of architecture as the Host and does not need to be mentioned separately.

Normally, the agents are deployed on critical or high risks servers as a mean to keep deployment and maintenance cost under control. However, deployment and maintenance of the software agent will prove to be a major challenge in enterprise that lacks a centralized software deployment mechanism. Often, this means dedicating resource to physically be onsite in order to install, upgrade, or update the agent software.

In addition, connectivity challenges similar to those mentioned in the NIDS architecture need to be considered. It is especially true in this case as most H/Hybrid IDS deployment utilizes the existing network to communicate.

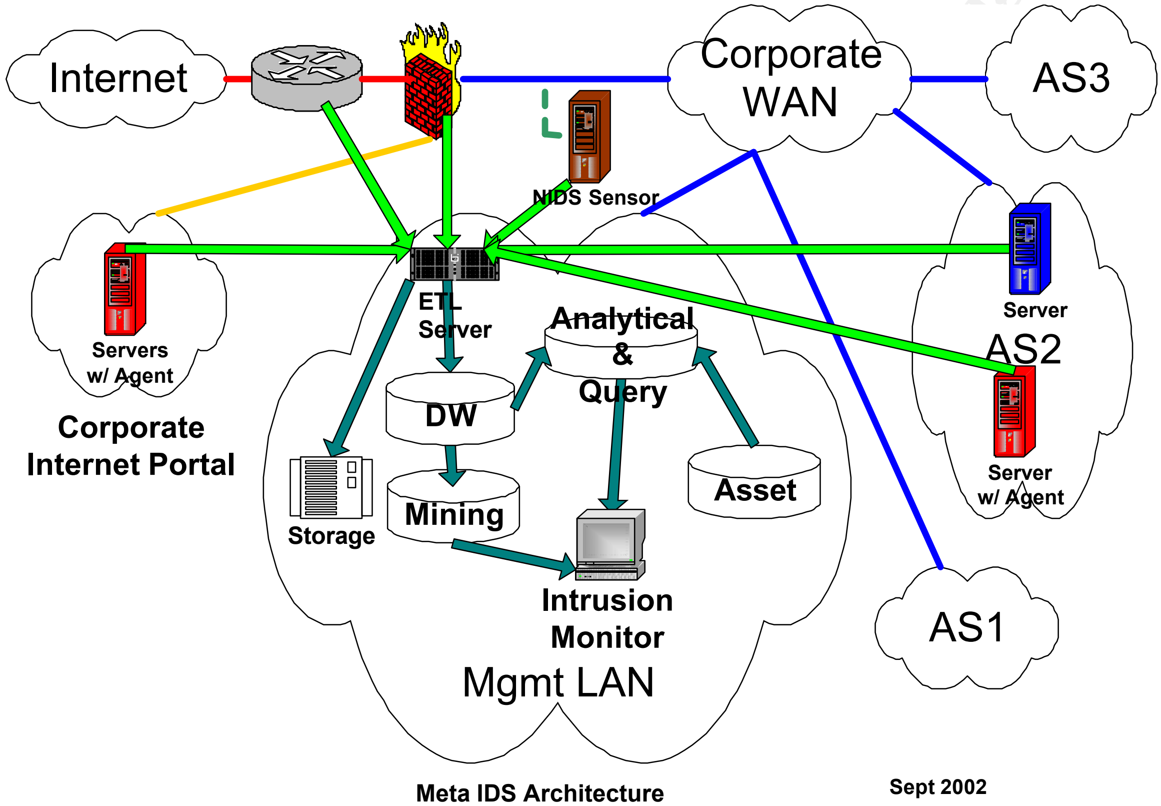


<u>Meta IDS</u>

In the diagram below, the components of Meta IDS are broken down into individual pieces (as shown in the management network segment) to illustrate the various stages required to complete the full deployment. However, the number of machines required can be reduced by having powerful machines serving a combination of functions. One key point to keep in mind is that performance monitoring is mandatory in order to balance cost versus performance.

The Extract Transform and Load (ETL) server illustrated below is the staging area where aggregation and normalization occurs. This is where logs and alerts data from sources such as servers, NIDS sensors, HIDS sensors, routers, and firewalls would be collected, sanitized and pushed onto the data warehouse. Optionally, the raw data from the various devices can be store separately if forensic is a requirement. An analysis database, with a refresh rate of approximately one week worth of top fill data extracted from the data warehouse, complete with customized rules can be utilized to perform real-time query and analytic to determine and raise intrusion alerts to the intrusion monitor. The rules can consist of comparison against known attack pattern, check for anomalies, check for threshold violation, and correlation of data against other devices, correlation of findings against existing states of the targeted assets. Long term trending and analysis can performed via the a mining database, with a refresh rate of approximately 6 months worth of top fill data extracted from the data warehouse, complete with a different set of customized rules. This enable the correlation of events over long term and build a comprehensive picture of what is happening and is especially useful in detecting the slow and methodical attacks.

The Intrusion Monitor, in addition to visual alerting, will also work to generate reports, and as a notification manager if remote alerting is a requirement. The Asset database, in order for the process of eliminating false positives to be automated, will need to contains detailed information on the targeted device, such as platforms, patch level, active applications, network services, and even vulnerability information from vulnerability assessment results.



Process

Once the IDS infrastructure is in place, attention must be paid to processes required to complement the technology gaps. These processes can be shared among the various IDS technology with relatively minor variations. Keep in mind that special consideration must be given to IDS equipments that resides in graphically diverse locations. Chances are these locations do not have dedicated and skilled staffs available to carry out some of the necessary work.

<u>Configuration</u>

After deployment
Out of the box, the default IDS settings will generate a lot of false positive and will require a lot of care and attentions. Fine-tuning of the IDS sensors will be a breeze if there exists

an asset database that contains up-to-date information pertinent to the servers and its operation. Information such as OS platforms, software applications, and software patch levels, network services, relationship to other hosts, owners/contact info, and host vulnerability from VA results. The information is vital to the elimination of false-positives and hence resources cost associated with investigating intrusion attempts.

Accommodating changes
Overtime, with sufficient care and investigation the amount of false positive will be cut down to a minimum, however, the IDS will required re-configurations to suit the environmental change associated with new vulnerabilities, threats, business needs, and technology changes. A process requiring documentation of the changes and periodic review of such changes will payback in the long term. Often, this can be accomplished via existing change management process in the enterprise.

Maintenance

Hardware
The fact of life is machine do break down or upgrade is required to meet processing demand and should be taken into account when planning for acquisition and resources. Periodic review of the need and support contracts will ensure a smooth operation. As the operation age, this need will become even more apparent.

Software
Software will need to be updated regularly, whether it's the OS, its patches, the IDS product, its patches, its updates, or the supporting applications and their patches and updates. Having a process in place to periodic review and maintain the IDS software up to date will ensure that the operation remains effective.

Monitoring

Alerting
Without monitoring, the IDS equipments are nothing more than expensive pieces of junk. In order to be effective, the IDS console needs to be monitored for alerts and the level of effectiveness is directly related to the amount of monitoring taking place. Monitoring can be accomplish via 24x7x365 onsite staffs, scheduled check of the console, or virtual alerting via mechanism such as SNMP trap, email, pager, syslog, secured web portal, SMS, or RIM.

Intrusion Response
Having said all the above, the next step in ensuring IDS operation effectiveness is to have a response process that support the alerts raised by monitoring. If passive detection is the

objective set out in deploying the IDS in the enterprise, then next step is to make note of the alerts and compile a report based on the information received. However, in most enterprise, the objective is more comprehensive than just reporting in order to justify for the costly investment associated with this type of deployment. Having processes defined for responding to the various intrusion alerts will ensure that the deliverables are inline with the set objective of IDS deployment. In most enterprise, this entails adding a provision for responding to IDS alerts to the existing intrusion response plan. In addition, a tracking mechanism integrated as part of this process will have many benefits including that of detecting the slow and methodical attacks, and metric to measure the IDS effectiveness level.

Incident Handling
In the event that the intrusion was successful, there is a need to manage the incident so that the damage is minimized. An established incident handling guidelines should be in place to facilitate an effective response to the emergency. This might entail revisiting the exiting incident handling plan and make specific provision for handling IDS reported incidents.

Reporting
Statistical reporting is one of the key measures of IDS deployment effectiveness and over time, the trend associated with these data can provide a measure of risks in the enterprise. Having a process in place, whether automated or manual, to regularly report on events, attacks, corrective actions, and performance will ensure that a balance can be achieved between cost and effectiveness.

Resources

Hardware

There must be a budget in place to address the problem associated volume management, maintenance, and upgrades. Depending on the size of deployment and the enterprise policy on retention, data growth is expected to be in the terabyte. In addition, extra fund will be required if redundancy is also defined as a requirement.

Software

Budget should be allocated to not only OS and IDS software and their maintenance but also supporting software required to analyze, track, and report. Over time, these will need to be upgraded in order to keep up with the constant changes.

Facility

In order to have an effective IDS monitoring functionality in an enterprise network, budget should be allocated to creating a security operation center. Integration with existing NOC might be an alternative if security issues such as confidentiality and integrity of the investigation are properly addressed.

Personnel

Depending on the intrusion response plan, the need for staffs will vary based on the scheme utilized. However, it should be noted that the IDS effectiveness level is directly related to the skill level of the IDS analysts and whether the analysts have enough time to carry out a thorough investigation. Often, this is the second sacrificial lamb, after training, in the IDS design when it comes to cost cutting. The danger is the false sense of security in thinking that the IDS system is still performing at the same effectiveness level with minimal supervision. The overload problem is a serious one as a compromise with evidence erasure can be completed in minutes if not seconds. The outcome of an attack that was not addressed is the same regardless if the cause of lack of response is due to ignorance or incompetence.

Fine-tuning
Initial configuration of IDS after deployment will required a lot of investigation, development, tuning, and testing. Budget should be allocated to ensure that this kind of optimization takes place in order to ensure that IDS operation effectiveness. The long-term payoff is substantial, as cost associated with investigating false positives will be brought to a manageable level.

Skills
Ideally, in order to be fully effective, the current IDS analyst should at least posses the following core skills:

- Network including design and communication protocols (packet level)
- Programming algorithm and some high level languages such as C
- Database implementation and usage
- Unix and/or Windows operating system (advanced level)
- Communication

IDS analyst with experience is more valuable than less experienced one as they are more exposed to the variety of attacks and would be in a better position to recognize the patterns.

Training
Keeping up to date with new vulnerabilities and threats must be a priority for the IDS analysts in order to sustain the same level of effectiveness. As such, budget should be set aside to allow IDS analysts to update and upgrade their skills. Alternative would be to provide them with sufficient time to perform daily research. However, in most organization, this is seen as unproductive as the benefits are difficult to measure.

References

Goeldenitz, Thomas. "IDS – Today and Tomorrow". January 22, 2002. URL: http://rr.sans.org/intrusion/today.php (September 12, 2002).

Kinn, David and Timm, Kevin. "Justifying the expense of IDS, Part One: An Overview of ROIs for IDS". July 18, 2002. URL: http://online.securityfocus.com/infocus/1608 (September 12, 2002).

Kinn, David and Timm, Kevin. "Justifying the expense of IDS, Part Two: An Overview of ROIs for IDS". August 27, 2002. URL: http://online.securityfocus.com/infocus/1621 (September 12, 2002).

Kothari, Pravin. "Intrusion Detection Interoperability and Standardization". February 19, 2002. URL: http://rr.sans.org/intrusion/interop.php (September 12, 2002).

Laing, Brian. "How To Guide: Implementing a Network Based Intrusion Detection System". 2000. URL: http://www.snort.org/docs/iss-placement.pdf (September 12, 2002).

Lee, Stolfo, Chan, Eskin, Fan, Miller, Hershkop, Zhang. "Real Time Data Mining-based Intrusion Detection". June 2001. URL: http://www.cs.columbia.edu/ids/publications/dmids-discex01.pdf (September 12, 2002).

Lee, Stolfo, Mok. "A Data Mining Framework for Building Intrusion Detection Models". May 1999. URL: http://www.cs.columbia.edu/ids/publications/wenke-ieee99.ps (September 12, 2002)

Loshin, Pete. "Meta Detection". Information Security Magazine. August 2001. URL: http://www.infosecuritymag.com/articles/august01/cover.shtml (September 12, 2002).

Morris, Chris. "What Do You Do After You Deploy the IDS?". January 3, 2001. URL: http://www.sans.org/newlook/resources/IDFAQ/deploy.htm (September 12, 2002).

Northcutt, Stephen. "What is network based intrusion detection?" Intrusion Detection FAQ Version 1.52. URL: http://www.sans.org/newlook/resources/IDFAQ/network_based.htm (September 12, 2002).

Scott, J. Steven. "Threat Management: The State of Intrusion Detection". August 9, 2002. URL: http://www.snort.org/docs/threatmanagement.pdf (September 12, 2002).

Shah, Baiju. "How to Choose Intrusion Detection Solution". July 24, 2001. URL:

http://rr.sans.org/intrusion/choose.php (September 12, 2002)

Zirkle, Laurie. "What is host-based intrusion detection?" Intrusion Detection FAQ Version 1.52. URL: http://www.sans.org/newlook/resources/IDFAQ/host_based.htm (September 12, 2002).

Assignment #2: Three Network Detects

Detect #1:

1) Source of trace:

http://www.incidents.org/logs/Raw/2002.5.30 and is in standard tcpdump format.

2) Detect was generated by:

Tcpdump binary log was generated by Snort ruleset. The first 13 packets triggered the "WEB-IIS .... access" rule and the last 2 packets triggered the "WEB-IIS cmd.exe access". Analysis was performed with Snort 1.8.7 with default ruleset, Ethereal, and Tcpdump/Windump.

```
>tcpdump -n -v -r 2002.5.30 ip host 4.63.141.232

20:09:03.494488 IP (tos 0x0, ttl 110, id 51666, len 136) 4.63.141.232.3588 >
46.5.180.151.80: P [bad tcp cksum b3fd (->bfba)!] 830374698:830374794(96) ack
4183985374 win 64240 (DF)bad cksum d3df (->cdd9)!
20:09:03.664488 IP (tos 0x10, ttl 240, id 0, len 135) 4.63.141.232.3588 > 46.5.180.151.80: P
[bad tcp cksum 0 (->4098)!] 96:191(95) ack 110981923 win 0bad cksum 0 (->559d)!
20:09:03.734488 IP (tos 0x0, ttl 110, id 51731, len 157) 4.63.141.232.3598 >
46.5.180.151.80: P [bad tcp cksum 4ae8 (->aac3)!] 830930012:830930129(117) ack
4180460333 win 64240 (DF)bad cksum d389 (->cd83)!
20:09:03.954488 IP (tos 0x10, ttl 240, id 0, len 156) 4.63.141.232.3598 > 46.5.180.151.80: P
[bad tcp cksum 0 (->2c7)!] 117:233(116) ack 139699240 win 0bad cksum 0 (->5588)!
20:09:04.044488 IP (tos 0x0, ttl 110, id 51808, len 157) 4.63.141.232.3599 >
46.5.180.151.80: P [bad tcp cksum 890 (->686b)!] 831023947:831024064(117) ack
4192574186 win 64240 (DF)bad cksum d33c (->cd36)!
20:09:04.214488 IP (tos 0x10, ttl 240, id 0, len 156) 4.63.141.232.3599 > 46.5.180.151.80: P
[bad tcp cksum 0 (->1b9)!] 117:233(116) ack 102393111 win 0bad cksum 0 (->5588)!
20:09:04.284488 IP (tos 0x0, ttl 110, id 52224, len 185) 4.63.141.232.3602 >
46.5.180.151.80: P [bad tcp cksum ba09 (->f35f)!] 831258990:831259135(145) ack
4188403520 win 64240 (DF)bad cksum d180 (->cb7a)!
20:09:04.464488 IP (tos 0x10, ttl 240, id 0, len 184) 4.63.141.232.3602 > 46.5.180.151.80: P
[bad tcp cksum 0 (->6e12)!] 145:289(144) ack 1096763473 win 0bad cksum 0 (->556c)!
20:09:05.564488 IP (tos 0x0, ttl 110, id 61701, len 138) 4.63.141.232.3623 >
46.5.180.151.80: P [bad tcp cksum b62e (->817a)!] 832504296:832504394(98) ack
4188201590 win 64240 (DF)bad cksum acaa (->a6a4)!
20:09:05.704488 IP (tos 0x0, ttl 110, id 61976, len 136) 4.63.141.232.3630 >
46.5.180.151.80: P [bad tcp cksum f27 (->de2)!] 832886489:832886585(96) ack
```

4186565264 win 64240 (DF)bad cksum ab99 (->a593)!
20:09:05.764488 IP (tos 0x10, ttl 240, id 0, len 135) 4.63.141.232.3630 > 46.5.180.151.80: P [bad tcp cksum 0 (->4c89)!] 96:191(95) ack 1101232723 win 0bad cksum 0 (->559d)!
20:09:05.854488 IP (tos 0x0, ttl 110, id 62165, len 140) 4.63.141.232.3632 > 46.5.180.151.80: P [bad tcp cksum 91b4 (->63d2)!] 833018479:833018579(100) ack 4185390156 win 64240 (DF)bad cksum aad8 (->a4d2)!
20:09:06.014488 IP (tos 0x10, ttl 240, id 0, len 139) 4.63.141.232.3632 > 46.5.180.151.80: P [bad tcp cksum 0 (->b323)!] 100:199(99) ack 1102408083 win 0bad cksum 0 (->5599)!
20:09:06.084488 IP (tos 0x0, ttl 110, id 62537, len 136) 4.63.141.232.3640 > 46.5.180.151.80: P [bad tcp cksum d4f (->190c)!] 833436387:833436483(96) ack 4184964703 win 64240 (DF)bad cksum a968 (->a362)!
20:09:06.264488 IP (tos 0x10, ttl 240, id 0, len 135) 4.63.141.232.3640 > 46.5.180.151.80: P [bad tcp cksum 0 (->e9b0)!] 96:191(95) ack 1102833725 win 0bad cksum 0 (->559d)!

3) Probability the source address was spoofed:

Based on the amount of data obtained, the source address has a high probability of being spoofed since the envelope has been tempered with as showcased by the header checksum. More details to follow in the Attack Mechanism section.

4) Description of the attack:

The attack was based on "File Permission Canonicalization" Vulnerability and "Web Server Folder Traversal" Vulnerability of Microsoft IIS 4/5. However, the packets have been mangled and the attack codes have been modified such that it will trigger IDS systems rather than achieve any real results for the attacker.

Further information on CVE-2000-0884 is available at http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2000-0884 and at
In addition on CVE-2001-0333 is available at http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2001-0333

http://www.kb.cert.org/vuls/id/111677

5) Attack mechanism:

The attack with modified probe sequences based on the Nimda and Code Red is performed via an automated script (all 15 packets were received in 2.77 seconds) with some scripting errors. However, there was no attempt to probe for the existence of root.exe and all 15 attempts were directed at cmd.exe.

Looking at the first two packets that triggered the "WEB-IIS .... access" Snort alert:

06/29-20:09:03.494488 4.63.141.232:3588 -> 46.5.180.151:80

TCP TTL:110 TOS:0x0 ID:51666 IpLen:20 DgmLen:136 DF
AP Seq: 0x317E832A Ack: 0xF9628CDE Win: 0xFAF0 TcpLen: 20
=+=

06/29-20:09:03.664488 4.63.141.232:3588 -> 46.5.180.151:80
TCP TTL:240 TOS:0x10 ID:0 IpLen:20 DgmLen:135
AP Seq: 0x317E838A Ack: 0x0 Win: 0x0 TcpLen: 20
=+=

Note the differences between the two packets:

The first has TTL of 110, TOS of 0x0, ID of 51666, Datagram length of 136, Do not fragment flag set, Ack number 0xF9628CDE, and window size of 0xFAF0. This packet has the characteristics associated with Windows 2000/XP. However, the window size was changed to deliberately mislead the remote OS identification.

The second arrives 0.17 seconds after the first, using the same port, the same sequence number and has TTL of 240, TOS of 0x10, ID of 0, Datagram length of 135, Ack number 0x0, and window size of 0x0. This packet has the characteristics associated with Cisco IOS version 12.0. However, the window size and TOS were manipulated to deliberately mislead the remote OS identification

Information for passive fingerprint monitoring is available from
http://project.honeynet.org/papers/finger/traces.txt

Combined with the fact that the header checksum on both packets has been tampered with, there is a good chance that these packets have been crafted.

Looking at the payload on both packets:

 length = 94

000 : 47 45 54 20 2F 73 63 72 69 70 74 73 2F 2E 2E 5C   GET /scripts/..\
010 : 2E 2E 2F 77 69 6E 6E 74 2F 73 79 73 74 65 6D 33   ../winnt/system3
020 : 32 2F 63 6D 64 2E 65 78 65 3F 2F 63 2B 64 69 72   2/cmd.exe?/c+dir
030 : 20 72 20 72 20 48 54 54 50 2F 31 2E 30 0D 0A 48   r r HTTP/1.0..H
040 : 6F 73 74 3A 20 77 77 77 0D 0A 43 6F 6E 6E 6E 65   ost: www..Connne
050 : 63 74 69 6F 6E 3A 20 63 6C 6F 73 65 0D 0A   ction: close..

and

 length = 93

000 : 47 45 54 20 2F 73 63 72 69 70 74 73 2F 2E 2E 5C   GET /scripts/..\
010 : 2E 2E 2F 77 69 6E 6E 74 2F 73 79 73 74 65 6D 33   ../winnt/system3

020 : 32 2F 63 6D 64 2E 65 78 65 3F 2F 63 2B 64 69 72   2/cmd.exe?/c+dir
030 : 20 72 20 72 20 48 54 54 50 2F 31 2E 30 0D 0A 48   r r HTTP/1.0..H
040 : 6F 73 74 3A 20 77 77 77 0D 0A 43 6F 6E 6E 6E 65   ost: www..Connne
050 : 63 74 69 6F 6E 3A 20 63 6C 6F 73 65 0D   ction: close.

The difference in length size is due to the lack of 0A or newline character in the second packet.

The remaining 11 of 13 packets follow the similar patterns but with different payloads per pair. The significant changes to payload sizes are attributed to varying attempts. There is an exception at packet number 9 which did not have a packet pair.

The last two packets that triggered the "WEB-IIS cmd.exe access" Snort alert:

06/29-20:09:06.084488 4.63.141.232:3640 -> 46.5.180.151:80
TCP TTL:110 TOS:0x0 ID:62537 IpLen:20 DgmLen:136 DF
AP Seq: 0x31AD3AE3 Ack: 0xF9717E5F Win: 0xFAF0 TcpLen: 20
=+=

06/29-20:09:06.264488 4.63.141.232:3640 -> 46.5.180.151:80
TCP TTL:240 TOS:0x10 ID:0 IpLen:20 DgmLen:135
AP Seq: 0x31AD3B43 Ack: 0x3B2D669B Win: 0x0 TcpLen: 20
=+=

Again, similar modifications to the packets header along with bad checksum indicate that the packets are likely crafted.

Looking at the payload on both packets:

length = 94

000 : 47 45 54 20 2F 73 63 72 69 70 74 73 2F 2E 2E 2F   GET /scripts/../
010 : 2E 2E 2F 77 69 6E 6E 74 2F 73 79 73 74 65 6D 33   ../winnt/system3
020 : 32 2F 63 6D 64 2E 65 78 65 3F 2F 63 2B 64 69 72   2/cmd.exe?/c+dir
030 : 20 72 20 72 20 48 54 54 50 2F 31 2E 30 0D 0A 48   r r HTTP/1.0..H
040 : 6F 73 74 3A 20 77 77 77 0D 0A 43 6F 6E 6E 6E 65   ost: www..Connne
050 : 63 74 69 6F 6E 3A 20 63 6C 6F 73 65 0D 0A   ction: close..

and

length = 93

000 : 47 45 54 20 2F 73 63 72 69 70 74 73 2F 2E 2E 2F   GET /scripts/../
010 : 2E 2E 2F 77 69 6E 6E 74 2F 73 79 73 74 65 6D 33   ../winnt/system3

```
020 : 32 2F 63 6D 64 2E 65 78 65 3F 2F 63 2B 64 69 72   2/cmd.exe?/c+dir
030 : 20 72 20 72 20 48 54 54 50 2F 31 2E 30 0D 0A 48    r r HTTP/1.0..H
040 : 6F 73 74 3A 20 77 77 77 0D 0A 43 6F 6E 6E 6E 65   ost: www..Connne
050 : 63 74 69 6F 6E 3A 20 63 6C 6F 73 65 0D           ction: close.
```

Again the similarity is there, however, the "/" instead of "\" in the payload have triggered a different Snort rule.

It is plausible that the first packet in each payload pair is a valid TCP session (if the 3-way handshake packets were omitted from the capture for some reason) but not likely. However, the second pack in each payload-pair is not a genuine TCP packet. Crafted packets are usually rejected by host with up-to-date software.

Additionally, the payloads can be considered harmless. Attempts at directory listing and sometime with syntax errors are not as dangerous as attempts to trojan the machine. In addition, manipulation of the payload further our conclusion of intends.

The intends, it seems, is to confuse and insert false alerts to the NIDS so that other real and dangerous attack can take place beneath the radar.

6) Correlations:

IP address: 4.63.141.232
Hostname: tamqfl1-ar5-4-63-141-232.tamqfl1.dsl-verizon.net

Search result from ARIN for: ! NET-4-63-132-0-1 revealed

OrgName: GTE Intelligent Network Services
OrgID: GINS

NetRange: 4.63.132.0 - 4.63.151.255
CIDR: 4.63.132.0/22, 4.63.136.0/21, 4.63.144.0/21
NetName: GTEINS-63-132-30
NetHandle: NET-4-63-132-0-1
Parent: NET-4-0-0-0-1
NetType: Reassigned
Comment:
RegDate: 2002-05-01
Updated: 2002-05-01

TechHandle: VOH1-ARIN
TechName: Hostmaster, Verizon
TechPhone: +1-800-927-3000
TechEmail: hostmaster@bizmailsrvcs.net

OrgAbuseHandle: VOH1-ARIN
OrgAbuseName: Hostmaster, Verizon
OrgAbusePhone: +1-800-927-3000
OrgAbuseEmail: hostmaster@bizmailsrvcs.net

OrgNOCHandle: VOH1-ARIN
OrgNOCName: Hostmaster, Verizon
OrgNOCPhone: +1-800-927-3000
OrgNOCEmail: hostmaster@bizmailsrvcs.net

OrgTechHandle: VOH1-ARIN
OrgTechName: Hostmaster, Verizon
OrgTechPhone: +1-800-927-3000
OrgTechEmail: hostmaster@bizmailsrvcs.net

DShield Profile:
Country: US
Contact E-mail: csoulia@genuity.net
Total Records against IP: 12
Number of targets: 7
Date Range: 2002-07-02 to 2002-07-02
Ports Attacked (up to 10):
Port Attacks
Fightback: sent to csoulia@genuity.net on 2002-07-01 07:31:17
no reply received

Similar attack was analyzed in Michael Wilkinson's (GCIA Analyst #508) practical assignment. There also exists a possibility of defamation attack as this IP also has a record on DShield. Otherwise, the IP could belong to attacker.

Attacks based on Nimda and Code Red are very well documented and readily available at:

http://www.incidents.org/react/nimda.pdf and
http://www.incidents.org/react/code_redII.html

However, the probes seen here are based mainly on "File Permission Canonicalization" Vulnerability and "Web Server Folder Traversal" Vulnerability

http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/ms00-078.asp and
http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/ms00-057.asp

7) Evidence of active targeting:

This attack has specifically targeted the NIDS detecting intrusion for web server at 46.5.180.151. However, since there is no web server at 46.5.180.151, the attack is most likely random or the person running the script really do not know what they are doing or the person is fooling around or the machine at the attacker address (not the same as source shown since the packets are most likely spoofed) has been trojaned.

8) Severity:

Critically: 1 as there is no web server at the destination address.

Lethality: 3 as the dir command in the payload is not as critical as other possible commands.

System Countermeasures: 5 as there is no stronger measure than a non existence server.

Network Countermeasures: 5 as there is no stronger measure than a non existence/not Internet routable network.

Therefore, the severity ranking for these probes is:

-6 = (1+3) – (5+5)

As per the formula below

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

9) Defensive recommendation:

In this case, the defense mechanism can't be defeated. However, in normal circumstances where the target is real, the system should be fully patched in order to prevent exploits such as these from making any kind of impact.

Patch specific to this vulnerability can be obtained from the MS Bulletins listed under section 6) Correlations

10) Multiple choice test question:

GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir r HTTP/1.0
GET /_vti_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir c+dir HTTP/1.0
GET /_mem_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir c+dir HTTP/1.0

GET /msadc/..%5c../..%5c../..%5c/..55../..c1../../.../
winnt/system32/cmd.exe?/c+dir 32/cmd.exe?/c+dir HTTP/1.0
GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir dir HTTP/1.0
GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir r HTTP/1.0
GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir c+dir HTTP/1.0
GET /scripts/..%2f../winnt/system32/cmd.exe?/c+dir r HTTP/1.0

Which of the following attack best describes the above sequence?

1) CodeRedII Worm
2) Nimda Worm
3) Web Server Folder Traversal **Correct Answer**
4) Sadmind/IIS Worm

11) Posting to incidents.org:

Email header for posting to intrusions@incidents.org:

Date: Sat, 31 Aug 2002 14:15:09 -0700 (PDT)
From:"nsck" nsck2000@yahoo.com
Subject: LOGS: GIAC GCIA Version 3.3 Practical Detect(s) a
To:intrusions@incidents.org
Content-Length: 5005

Note: There was no URL available for the mailing list.

Top 3 Questions & Defenses:

None received.

12) Summary:

The probes are based mainly on "File Permission Canonicalization" Vulnerability and "Web Server Folder Traversal" Vulnerability. However, the packets and payload have been mangled such that the intent might be to distract the NIDS analyst so that other real and dangerous attack can take place beneath the radar.

Detect #2:

1) Source of trace:

http://www.incidents.org/logs/Raw/2002.5.9 and is in standard tcpdump format.

2) Detect was generated by:

Tcpdump binary log was generated by Snort ruleset. The 9 packets triggered the "SCAN SOCKS Proxy attempt" rule. Analysis was performed with Snort 1.8.7 with default ruleset, Ethereal, and Tcpdump/Windump.

```
>tcpdump -n -v -r 2002.5.9 ip host 67.113.244.112

14:51:50.154488 IP (tos 0x0, ttl 111, id 20084, len 48) 67.113.244.112.65298 >
46.5.216.192.1080: S [bad tcp cksum 20d9 (->1ad3)!] 3919510722:3919510722(0) win
16384 <mss 1452,nop,nop,sackOK> (DF)bad cksum 84b2 (->7eac)!
14:51:53.144488 IP (tos 0x0, ttl 111, id 20327, len 48) 67.113.244.112.65298 >
46.5.216.192.1080: S [bad tcp cksum 20d9 (->1ad3)!] 3919510722:3919510722(0) win
16384 <mss 1452,nop,nop,sackOK> (DF)bad cksum 83bf (->7db9)!
14:51:59.144488 IP (tos 0x0, ttl 111, id 20815, len 48) 67.113.244.112.65298 >
46.5.216.192.1080: S [bad tcp cksum 20d9 (->1ad3)!] 3919510722:3919510722(0) win
16384 <mss 1452,nop,nop,sackOK> (DF)bad cksum 81d7 (->7bd1)!
16:10:41.614488 IP (tos 0x0, ttl 110, id 2211, len 48) 67.113.244.112.64933 >
46.5.157.228.1080: S [bad tcp cksum 9b90 (->958a)!] 2710701665:2710701665(0) win
16384 <mss 1452,nop,nop,sackOK> (DF)bad cksum 660 (->5a)!
16:10:44.604488 IP (tos 0x0, ttl 110, id 2443, len 48) 67.113.244.112.64933 >
46.5.157.228.1080: S [bad tcp cksum 9b90 (->958a)!] 2710701665:2710701665(0) win
16384 <mss 1452,nop,nop,sackOK> (DF)bad cksum 578 (->ff71)!
16:10:50.614488 IP (tos 0x0, ttl 110, id 2916, len 48) 67.113.244.112.64933 >
46.5.157.228.1080: S [bad tcp cksum 9b90 (->958a)!] 2710701665:2710701665(0) win
16384 <mss 1452,nop,nop,sackOK> (DF)bad cksum 39f (->fd98)!
16:22:57.514488 IP (tos 0x0, ttl 111, id 58726, len 48) 67.113.244.112.65248 >
46.5.88.71.1080: S [bad tcp cksum 16d2 (->fcb)!] 3182146665:3182146665(0) win 16384
<mss 1452,nop,nop,sackOK> (DF)bad cksum 6f3a (->6833)!
16:23:00.454488 IP (tos 0x0, ttl 111, id 58952, len 48) 67.113.244.112.65248 >
46.5.88.71.1080: S [bad tcp cksum 16d2 (->fcb)!] 3182146665:3182146665(0) win 16384
<mss 1452,nop,nop,sackOK> (DF)bad cksum 6e58 (->6751)!
16:23:06.464488 IP (tos 0x0, ttl 111, id 59426, len 48) 67.113.244.112.65248 >
46.5.88.71.1080: S [bad tcp cksum 16d2 (->fcb)!] 3182146665:3182146665(0) win 16384
<mss 1452,nop,nop,sackOK> (DF)bad cksum 6c7e (->6577)!
```

3) Probability the source address was spoofed:

Based on the amount of data obtained, the source address has a low to medium probability of being spoofed since the envelope has been tempered with as showcased by the header checksum. More details to follow in the Attack Mechanism section.

4) Description of the attack:

The attacker from IP address 67.113.244.112 is scanning for misconfigured proxy servers at the following IP address:

46.5.216.192
46.5.157.228
46.5.88.71

on port 1080 (typical SOCKS port). The scans consists of 3 SYN packets per host on port 1080, an ACK reply to the attacker would indicate a live proxy server. This type of reconnaissance is precursory to actual exploitation via a number of possible ways such as traffic redirection or hostile takeover.

There are no CVE associated with the proxy scan as it is normal TCP traffic. However, there are numerous CVE associated with exploitation of various proxy servers.

5) Attack mechanism:

The scans appeared to be automated as all 9 packets were received in repetitive time sequence.

Looking at the first three packets directed at 46.5.216.192 that triggered the Snort alert:

```
06/09-14:51:50.154488 67.113.244.112:65298 -> 46.5.216.192:1080
TCP TTL:111 TOS:0x0 ID:20084 IpLen:20 DgmLen:48 DF
******S* Seq: 0xE99EFCC2  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

06/09-14:51:53.144488 67.113.244.112:65298 -> 46.5.216.192:1080
TCP TTL:111 TOS:0x0 ID:20327 IpLen:20 DgmLen:48 DF
******S* Seq: 0xE99EFCC2  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

06/09-14:51:59.144488 67.113.244.112:65298 -> 46.5.216.192:1080
TCP TTL:111 TOS:0x0 ID:20815 IpLen:20 DgmLen:48 DF
******S* Seq: 0xE99EFCC2  Ack: 0x0  Win: 0x4000  TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK
```

Notice that they all carry the exact same sequence number 0xE99EFCC2 from the same port 65298. However the time sequence between the packets are approximately 3 and 6 seconds apart. The incremental IP ID indicates that they are 3 different packets.

Looking at the next three packets directed at 46.5.157.228 that triggered the Snort alert:

06/09-16:10:41.614488 67.113.244.112:64933 -> 46.5.157.228:1080
TCP TTL:110 TOS:0x0 ID:2211 IpLen:20 DgmLen:48 DF
******S* Seq: 0xA1920661 Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

06/09-16:10:44.604488 67.113.244.112:64933 -> 46.5.157.228:1080
TCP TTL:110 TOS:0x0 ID:2443 IpLen:20 DgmLen:48 DF
******S* Seq: 0xA1920661 Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

06/09-16:10:50.614488 67.113.244.112:64933 -> 46.5.157.228:1080
TCP TTL:110 TOS:0x0 ID:2916 IpLen:20 DgmLen:48 DF
******S* Seq: 0xA1920661 Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

The similarity is there, they all carry the exact same sequence number 0xA1920661 from the same port 64933. The time sequences between the packets are approximately 3 and 6 seconds apart and the incremental IP ID indicates that they are 3 different packets.

Looking at the last three packets directed at 46.5.88.71 that triggered the Snort alert:

06/09-16:22:57.514488 67.113.244.112:65248 -> 46.5.88.71:1080
TCP TTL:111 TOS:0x0 ID:58726 IpLen:20 DgmLen:48 DF
******S* Seq: 0xBDABB469 Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

06/09-16:23:00.454488 67.113.244.112:65248 -> 46.5.88.71:1080
TCP TTL:111 TOS:0x0 ID:58952 IpLen:20 DgmLen:48 DF
******S* Seq: 0xBDABB469 Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

06/09-16:23:06.464488 67.113.244.112:65248 -> 46.5.88.71:1080
TCP TTL:111 TOS:0x0 ID:59426 IpLen:20 DgmLen:48 DF
******S* Seq: 0xBDABB469 Ack: 0x0 Win: 0x4000 TcpLen: 28
TCP Options (4) => MSS: 1452 NOP NOP SackOK

Again, they all carry the exact same sequence number 0xBDABB469 from the same port 65248. The time sequences between the packets are approximately 3 and 6 seconds apart and the incremental IP ID indicates that they are 3 different packets.

Combined with the fact that the header checksum on all the packets has been tampered with, there is a chance that these packets have been crafted. Provided the TTL, TOS, DF, and Windows Size have not been tampered with, there is a very high probability that these packets came from a Windows 2000/XP machine.

Information for passive fingerprint monitoring is available from
http://project.honeynet.org/papers/finger/traces.txt

It is plausible that these packets came from a valid source if the 3-way handshake is present. However, it is also plausible that the packets came from a spoofed source.

The intends, it seems, is to confuse and insert false alerts to the NIDS so that other real and dangerous attack can take place beneath the radar or straight forward reconnaissance. Collectively, the data seems to indicate that it is the later.

6) Correlations:

IP address: 67.113.244.112
Hostname: adsl-67-113-244-112.dsl.snfc21.pacbell.net

Search results for: ! NET-67-113-244-0-1

CustName: PPPoX Pool Rback10 63.195.184.0
Address: 268 Bush St #5000 San Francisco, CA 94104
Country: US
Comment:
RegDate: 2002-06-21
Updated: 2002-06-21

NetRange: 67.113.244.0 - 67.113.244.255
CIDR: 67.113.244.0/24
NetName: SBCIS-062002160305
NetHandle: NET-67-113-244-0-1
Parent: NET-67-112-0-0-1
NetType: Reassigned
Comment:
RegDate: 2002-06-21
Updated: 2002-06-21

DShield Profile:
Country: US
Contact E-mail: abuse@pbi.net
Total Records against IP:
Number of targets:
Date Range: to
Ports Attacked (up to 10):
Port Attacks
Fightback: not sent

Similar attack was analyzed in Mark Embrich's (GCIA Analyst #491) practical assignment. There is no profile for this IP on DShield as this type of activity is considered non-intrusive and is likely categorized as noise by most organizations.

7) Evidence of active targeting:

The attacker has specifically targeted port 1080 for netblock 46.5.X.X.
However, since there is no proxy server at this netblock, the attack is most likely random or the person running the script really do not know what they are doing or the person is fooling around or the machine at the attacker address (not the same as source shown if the packets are spoofed) has been trojaned.

8) Severity:

Critically: 1 as there is no proxy server at the destination address.

Lethality: 2 as it is a random target but with specific port scan

System Countermeasures: 5 as there is no stronger measure than a non existence server.

Network Countermeasures: 5 as there is no stronger measure than a non existence/not Internet routable network.

Therefore, the severity ranking for these probes is:

-7 = (1+2) – (5+5)

As per the formula below

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

9) Defensive recommendation:

In this case, the defense mechanism can't be defeated. However, in normal circumstances where the target is real, proper configuration of the proxy server is required ensure further exploits from making an impact.

Additional information can be obtain from **http://help.undernet.org/proxyscan/**

10) Multiple choice test question:

What is the normal behaviour for a proxy server if you send it a SYN packet to TCP port 1080?

1) SYN/ACK packet reply from port 1080 **Correct Answer**
2) ACK packet reply from port 1080
3) ACK packet reply from port 113
4) SYN/ACK packet reply from port 113

11) Posting to incidents.org:

Email header for posting to intrusions@incidents.org:

Date: Sat, 31 Aug 2002 14:15:56 -0700 (PDT)
From:"nsck" nsck2000@yahoo.com
Subject: LOGS: GIAC GCIA Version 3.3 Practical Detect(s) b
To:intrusions@incidents.org
Content-Length: 3325

Note: There was no URL available for the mailing list.

Top 3 Questions & Defenses:

None received.

12) Summary:

The probes appear to be searching for proxy server running on port 1080 and can be categorized under reconnaissance.

Detect #3:

1) Source of trace:

http://www.incidents.org/logs/Raw/2002.6.4 and is in standard tcpdump format.

2) Detect was generated by:

Tcpdump binary logs were generated by Snort ruleset. The 16 packets triggered the "SCAN SYN FIN" rule. Analysis was performed with Snort 1.8.7 with default ruleset, Ethereal, and Tcpdump/Windump.

>tcpdump -n -v -r 2002.6.4 ip host 143.107.196.131

20:00:13.374488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.32.163.22:

SF [bad tcp cksum 9ba5 (->93a0)!] 1277370451:1277370451(0) win 1028bad cksum 773c (->6f37)!
20:24:07.204488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.224.95.22: SF [bad tcp cksum 2b66 (->265e)!] 1096280365:1096280365(0) win 1028bad cksum b482 (->af7a)!
21:07:22.194488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.70.106.22: SF [bad tcp cksum deae (->d7a7)!] 1856476573:1856476573(0) win 1028bad cksum 5077 (->4970)!
21:08:44.094488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.4.67.22: SF [bad tcp cksum acae (->a5a7)!] 548719713:548719713(0) win 1028bad cksum 929e (->8b97)!
21:20:42.384488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.33.195.22: SF [bad tcp cksum 7bf5 (->73f0)!] 1216478031:1216478031(0) win 1028bad cksum 761c (->6e17)!
21:25:12.344488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.18.8.22: SF [bad tcp cksum a0c8 (->99c1)!] 712092865:712092865(0) win 1028bad cksum 84d9 (->7dd2)!
21:27:40.234488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.105.234.22: SF [bad tcp cksum 68ac (->60a7)!] 785974883:785974883(0) win 1028bad cksum 2df5 (->25f0)!
21:29:03.344488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.66.214.22: SF [bad tcp cksum f97b (->f176)!] 246051027:246051027(0) win 1028bad cksum 5509 (->4d04)!
21:40:52.114488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.140.75.22: SF [bad tcp cksum 243 (->fd3a)!] 1519521051:1519521051(0) win 1028bad cksum 897 (->38f)!
22:26:24.554488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.71.200.22: SF [bad tcp cksum 8aed (->82e8)!] 345772237:345772237(0) win 1028bad cksum 5017 (->4812)!
23:02:27.934488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.152.53.22: SF [bad tcp cksum 218f (->1c87)!] 1187448314:1187448314(0) win 1028bad cksum fcac (->f7a4)!
23:02:58.244488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.184.138.22: SF [bad tcp cksum 7900 (->72fa)!] 1728050889:1728050889(0) win 1028bad cksum dd55 (->d74f)!
23:09:02.234488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.10.5.22: SF [bad tcp cksum 8d37 (->8630)!] 536451193:536451193(0) win 1028bad cksum 8cdc (->85d5)!
23:34:29.874488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.200.89.22: SF [bad tcp cksum f752 (->f24a)!] 1846110088:1846110088(0) win 1028bad cksum cc88 (->c780)!
00:00:50.184488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.213.21.22: SF [bad tcp cksum 4e (->fb45)!] 979991667:979991667(0) win 1028bad cksum bfcc (->bac4)!
00:05:46.854488 IP (tos 0x0, ttl 15, id 39426, len 40) 143.107.196.131.22 > 46.5.76.154.22:

SF [bad tcp cksum c4ce (->bcc9)!] 1934616395:1934616395(0) win 1028bad cksum 4b45 (->4340)!

3) Probability the source address was spoofed:

Based on the amount of data obtained, the source address has a low to medium probability of being spoofed since the envelope has been tempered with as showcased by the header checksum. More details to follow in the Attack Mechanism section.

4) Description of the attack:

The scan for a SSH server was crafted with the SYN/FIN flag. The attacker from IP address 143.107.196.131 is performing stealth scan for live SSH servers at the following IP address:

46.5.32.163
46.5.224.95
46.5.70.106
46.5.4.67
46.5.33.195
46.5.18.8
46.5.105.234
46.5.66.214
46.5.140.75
46.5.71.200
46.5.152.53
46.5.184.138
46.5.10.5
46.5.200.89
46.5.213.21
46.5.76.154

on port 22. The slow scans consists of 1 TCP packet with SYN and FIN flag set to port 22 of each of the target. A host with live SSH server would respond with either SYN/ACK (open port) or RST/ACK (closed port). This type of reconnaissance is precursory to actual exploitation.

There are no CVE associated with the scan but there are numerous CVE associated with SSH exploits.

5) Attack mechanism:

The scans appeared to be manually executed as all 16 packets were received in a period of approximately 2 hours with no repetitive time sequence.

Looking at the packets that triggered the Snort alert:

07/03-20:00:13.374488 143.107.196.131:22 -> 46.5.32.163:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x4C232053 Ack: 0x13F0F513 Win: 0x404 TcpLen: 20

07/03-20:24:07.204488 143.107.196.131:22 -> 46.5.224.95:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x4157E92D Ack: 0x40CBB7AF Win: 0x404 TcpLen: 20

07/03-21:07:22.194488 143.107.196.131:22 -> 46.5.70.106:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x6EA7959D Ack: 0x6CC19AA5 Win: 0x404 TcpLen: 20

07/03-21:08:44.094488 143.107.196.131:22 -> 46.5.4.67:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x20B4CC61 Ack: 0x5CE935D4 Win: 0x404 TcpLen: 20

07/03-21:20:42.384488 143.107.196.131:22 -> 46.5.33.195:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x4881FB4F Ack: 0x276E28CB Win: 0x404 TcpLen: 20

07/03-21:25:12.344488 143.107.196.131:22 -> 46.5.18.8:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x2A71ACC1 Ack: 0x2BC37AFE Win: 0x404 TcpLen: 20

07/03-21:27:40.234488 143.107.196.131:22 -> 46.5.105.234:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x2ED90663 Ack: 0x7C4BADA4 Win: 0x404 TcpLen: 20

07/03-21:29:03.344488 143.107.196.131:22 -> 46.5.66.214:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0xEAA70D3 Ack: 0x614C14A7 Win: 0x404 TcpLen: 20

07/03-21:40:52.114488 143.107.196.131:22 -> 46.5.140.75:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x5A920D1B Ack: 0x10AF27DB Win: 0x404 TcpLen: 20

07/03-22:26:24.554488 143.107.196.131:22 -> 46.5.71.200:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x149C10CD Ack: 0x8D730CD Win: 0x404 TcpLen: 20

07/03-23:02:27.934488 143.107.196.131:22 -> 46.5.152.53:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40

******SF Seq: 0x46C705FA Ack: 0x3079F7C6 Win: 0x404 TcpLen: 20

07/03-23:02:58.244488 143.107.196.131:22 -> 46.5.184.138:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x66FFF6C9 Ack: 0x62F13D7E Win: 0x404 TcpLen: 20

07/03-23:09:02.234488 143.107.196.131:22 -> 46.5.10.5:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x1FF99879 Ack: 0x504D90C8 Win: 0x404 TcpLen: 20

07/03-23:34:29.874488 143.107.196.131:22 -> 46.5.200.89:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x6E096788 Ack: 0x67FD318A Win: 0x404 TcpLen: 20

07/04-00:00:50.184488 143.107.196.131:22 -> 46.5.213.21:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x3A697C73 Ack: 0x30F37192 Win: 0x404 TcpLen: 20

07/04-00:05:46.854488 143.107.196.131:22 -> 46.5.76.154:22
TCP TTL:15 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x734FE74B Ack: 0x49637C5B Win: 0x404 TcpLen: 20

Notice that they all carry the exact same source port of 22, TTL of 15, TOS of 0x0, IP ID of 39426, IP length of 20, Datagram length of 40, Window size of 0x404, and TCP length of 20. This type of characteristic is consistent with the behaviour of Synscan tool or its variant.

Combined with the fact that the header checksum on all the packets has been tampered with, there is a very high probability that these packets have been crafted.

It is plausible that these packets came from a valid source if the 3-way handshake is present. However, it is also plausible that the packets came from a spoofed source. The attacker took measure to avoid detection by performing a slow and stealth scan. As such, it is more likely that the intent is to find live SSH servers. This means that the source IP address is most likely to be valid.

6) Correlations:

IP address: 143.107.196.131
Hostname: serpat.fmrp.usp.br

Search results for: 143.107.196.131

OrgName: Universidade de Sao Paulo

OrgID: UDSP

NetRange: 143.107.0.0 - 143.107.255.255
CIDR: 143.107.0.0/16
NetName: USP-ANSP
NetHandle: NET-143-107-0-0-1
Parent: NET-143-0-0-0-0
NetType: Direct Assignment
NameServer: BEE.USPNET.USP.BR
NameServer: BEE08.USPNET.USP.BR
Comment:
RegDate: 1990-03-26
Updated: 2002-04-15

TechHandle: ES788-ARIN
TechName: Santos Moreira, Edson
TechPhone: +55-11-3091-6328
TechEmail: cceadmin@usp.br

DShield Profile:
Country: BR
Contact E-mail: root@cce.usp.br
Total Records against IP: 71
Number of targets: 56
Date Range: 2002-07-03 to 2002-07-04
Ports Attacked (up to 10):
Port Attacks
Fightback: sent to root@cce.usp.br on 2002-07-03 22:50:05
no reply received

Similar attack was analyzed in Jalal Moloo's (GCIA Analyst #496) practical assignment.

It should be noted that this IP has a high number of records against it on DShield, which makes it a good candidate to be put on a "watch" list.

7) Evidence of active targeting:

The attacker has specifically targeted port 22 for netblock 46.5.X.X.
However, since there is no SSH server at this netblock, the attack is most likely random or the person running the script really do not know what they are doing or the person is fooling around or the machine at the attacker address (not the same as source if the packets are spoofed) has been trojaned.

8) Severity:

Critically: 1 as there is no SSH server at the destination address.

Lethality: 2 as it is a random target but with specific port scan

System Countermeasures: 5 as there is no stronger measure than a non existence server.

Network Countermeasures: 5 as there is no stronger measure than a non existence/not Internet routable network.

Therefore, the severity ranking for these probes is:

-7 = (1+2) – (5+5)

As per the formula below

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

9) Defensive recommendation:

In this case, the defense mechanism can't be defeated. However, in normal circumstances where the target is real, having updated versions of sshd is strongly recommended. In addition, filtering rules to limit IP from accessing the SSH server will be greatly beneficial.

10) Multiple choice test question:

What is the normal behaviour for SSH server it you send it a SYN/FIN packet on TCP port 22?

1) PSH/ACK packet reply
2) RST/ACK packet reply
3) FIN/ACK packet reply
4) SYN/ACK packet reply **Correct Answer**

11) Posting to incidents.org:

Email header for posting to intrusions@incidents.org:

Date: Sat, 31 Aug 2002 14:16:36 -0700 (PDT)
From:"nsck" nsck2000@yahoo.com
Subject: LOGS: GIAC GCIA Version 3.3 Practical Detect(s) c
To:intrusions@incidents.org

Content-Length: 3975

Note: There was no URL available for the mailing list.

Top 3 Questions & Defenses:

1) ID=39426 corresponds to a specific worm/scanner. So guess why you saw this and what it meant? <donald.smith@qwest.com>

This appears to be a customized version of SynScan tool. The TTL, Windows size and IP ID are consistent and this was one of the characteristics of SynScan.

2) Does SynScan always send a SYN after its initial syn/fin? If not under what conditions does it send a syn to gather the version information? <donald.smith@qwest.com>

The proper behaviour for SynScan is to send a SYN only after it receive a SYN/ACK.

3) Can you run SynScan with delays to get it under the radar of the ids?

Yes, you can specify the delay depends on your connection speed. However, the timing between packets seems to indicate that the attack was not automated.

Here are all the packets and their time deltas (don't think we need to go down to seconds at this point yet):

First one at 20:00:13 to 46.5.32.163
next one is 24 mins from previous to 46.5.224.95
next one is 41 mins from previous to 46.5.70.106
next one is 1 min from previous to 46.5.4.67
next one is 12 mins from previous to 46.5.33.195
next one is 5 mins from previous to 46.5.18.8
next one is 2 mins from previous to 46.5.105.234
next one is 2 mins from previous to 46.5.66.214
next .. 11 mins .. 46.5.140.75
next .. 46 mins .. 46.5.71.200
next .. 36 mins .. 46.5.152.53
next .. 30 secs .. 46.5.184.138
next .. 7 mins .. 46.5.10.5
next .. 25 mins .. 46.5.200.89
next .. 26 mins .. 46.5.213.21
last .. 5 mins .. 46.5.76.154

There is no repetitive time sequence over the 2 hours. Looking at the smallest delta: 30 secs. Enough time for most people to execute a single command line with options.

12) Summary:

The stealth and manual scan, based on a modified SynScan tool, appears to be searching for SSH server running on port 22.

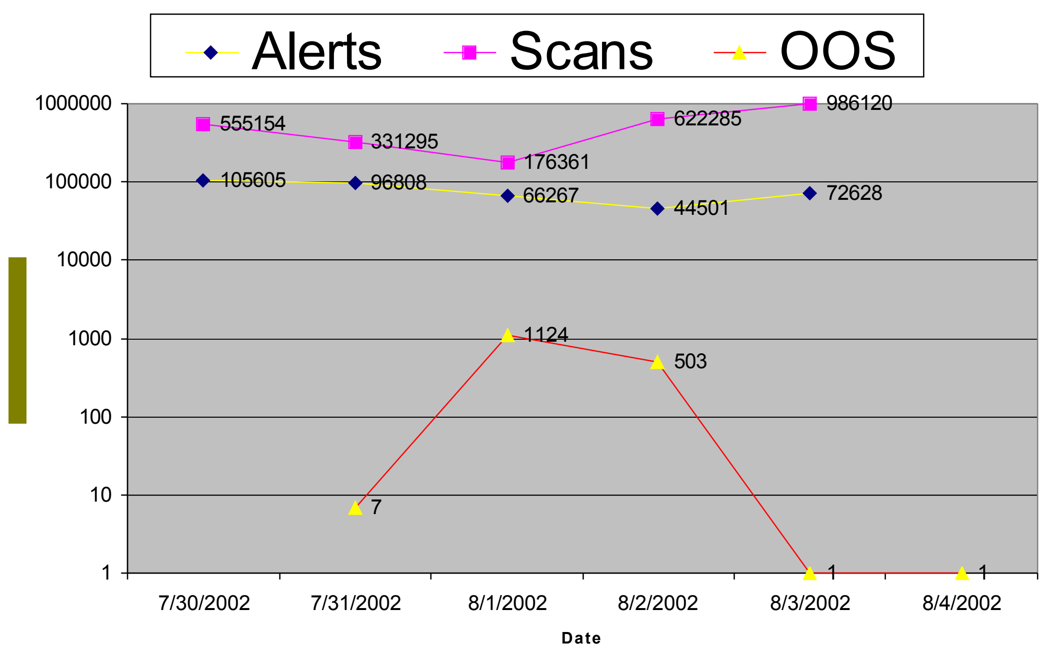
Assignment #3: Analyze This

Executive Summary

The audit for the period of July 30th through August 4th was focused on the analysis of alerts, scans, and out of specification (OOS) data provided by the University. Highlights:

- 385,809 alert events for the period of July 30th through August 3rd.
- 2,671,215 scan events for the period of July 30th through August 3rd.
- 1,636 OOS events for the period of July 31st through August 4th.

The graph below illustrates the trends associated with the events type. Events associated with Alerts and Scans seems to correspond throughout the week. However, events associated with OOS seem to contradict Alert and Scans. This is not normal and as such, the analysis might be skewed based on the quality of the data provided. Additional value could be added to this audit if it was performed against the University's security policy, and that infrastructure network diagram and the IDS sensor's placement information were available.



Analysis of the data provided that the University computers were scanned and attacked. However, there is insufficient evidence to say if they were compromised or not. Further follow up is required as detailed in the Conclusion and Defensive Recommendations section.

Logs Analyzed

Five days worth of log data accumulated by one or more Snort Intrusion Detection System sensor(s) strategically located around the University were collected for the audit. Three sets of log files running from July 30th through August 5th of 2002, were used in the analysis and are detailed as below:

Alert	*Size*
alert.020730.gz	1,538,518
alert.020731.gz	1,247,307
alert.020801.gz	844,437
alert.020802.gz	1,069,475
alert.020803.gz	1,150,676

The data from all these files were combined into a single file for analysis. In addition, preprocessing on the combined data was performed to eliminate duplicate data available in the raw scan data (below). Details on how it was done are documented in Analysis Process near the end of the document.

Scans	*Size*
scans.020730.gz	3,934,492
scans.020731.gz	2,202,003
scans.020801.gz	1,344,265
scans.020802.gz	4,391,619
scans.020803.gz	6,595,155

Again, the data from the scans files were combined into a single file for analysis. Similarly, the data from the Out of Spec (OOS) files were also combined into a single file for analysis. Data captured in the OOS log files pertained to strange or non-RFC complaint packets. Please note due to hardware failures, OOS log files for July 30th and 31st of 2002 were not available. However, the University staffs were kind enough to provide the OOS log files for August 4th and 5th of 2002 as alternatives.

OOS	*Size*
oos_Aug.1.2002.gz	544
oos_Aug.2.2002.gz	35,863
oos_Aug.3.2002.gz	17,080
oos_Aug.4.2002.gz	205
oos_Aug.5.2002.gz	194

Alerts Data Analysis

A total of 385,809 events were captured and generated 54 different types of alerts between July 30th and August 3rd. The breakdown based on dates are as followed:

```
| 105605   Jul/30
|
| 96808    Jul/31
|
| 72628    Aug/03
|
| 66267    Aug/01
|
| 44501    Aug/02
|
```

The chart below breakdown the events by alert type:

| 182974 UDP SRC and DST outside network
| 68258 spp_http_decode: IIS Unicode attack detected
| 36635 spp_http_decode: CGI Null Byte attack detected
| 26387 SMB Name Wildcard
| 23284 Watchlist 000220 IL-ISDNNET-990517
| 19799 TFTP - External UDP connection to internal tftp server
| 12145 External RPC call
| 3256 Possible trojan server activity
| 2483 SUNRPC highport access!
| 1658 IRC evil - running XDCC
| 1353 SNMP public access
| 1347 Null scan!
| 1258 Watchlist 000222 NET-NCFC
| 969 Queso fingerprint
| 679 Samba client access
| 677 Attempted Sun RPC high port access
| 549 Incomplete Packet Fragments Discarded
| 490 High port 65535 udp - possible Red Worm - traffic
| 324 NMAP TCP ping!
| 199 EXPLOIT x86 NOOP
| 124 Tiny Fragments - Possible Hostile Activity
| 121 beetle.ucs
| 105 SMB C access
| 77 EXPLOIT x86 setuid 0
| 63 ICMP SRC and DST outside network
| 58 EXPLOIT x86 stealth noop
| 57 STATDX UDP attack
| 38 IDS552/web-iis_IIS ISAPI Overflow ida nosize

| 36 EXPLOIT x86 setgid 0
| 31 TFTP - Internal UDP connection to external tftp server
| 24 Probable NMAP fingerprint attempt
| 19 HelpDesk 130.85.70.50 to External FTP
| 18 SYN-FIN scan!
| 16 connect to 515 from outside
| 16 High port 65535 tcp - possible Red Worm - traffic
| 9 External FTP to HelpDesk 130.85.70.49
| 9 TCP SRC and DST outside network
| 9 External FTP to HelpDesk 130.85.70.50
| 7 HelpDesk 130.85.70.49 to External FTP
| 7 HelpDesk 130.85.83.197 to External FTP
| 5 DDOS shaft client to handler
| 4 RFB - Possible WinVNC - 010708-1
| 3 NIMDA - Attempt to execute cmd from campus host
| 2 PHF attempt
| 2 Traffic from port 53 to port 123
| 2 SMB CD...
| 2 tw33dl3
| 2 connect to 515 from inside
| 2 130.85.30.3 activity
| 1 130.85.30.4 activity
| 1 Back Orifice
| 1 External FTP to HelpDesk 130.85.83.197

As seen in the table above, the majority of the events were raised by the first seven alert types captured. These are possibly generated by Intrusion Detection System that are poorly configured and/or were not poorly deployed. Further exploration of these events is required to determine the possible cause.

Assumption: MY.NET is the prefix for the University's network.

Frequent Alert Details (Generated more than 10,000 events)

UDP SRC and DST outside network
Severity: Noise Reported: 182,974 times

These UDP packets were captured by the Snort IDS sensor(s) did not appear to have originated from or destined to University's network.

Although there are a number of source addresses, there are only 3 destination IP addresses and 2 ports. The 3 IP addresses are in the class D address space that is reserved for multicast applications. The two major IP source 63.250.213.12 and 63.250.213.73 are registered to Yahoo!Broadcast. It appears that 233.28.65.148 and 233.28.65.173 are multicast clients and 233.2.171.1 is a multicast server.

Additionally, there are lots of NetBIOS traffics coming from 3.0.0.99 (General Electric) to a private 10.0.0.1 address. If the 10.0.0.1 address is an internal University's address, then verdict is at the opposite end of spectrum.

Correlations: Scott Shinberg (GCIA Analyst #389) noted this event in his practical assignment and found similar results.

Recommendations: Alter the Snort rule so that multicast traffic will not trigger this alert. Contact General Electric and inform them that their network is leaking NetBIOS.

spp_http_decode: IIS Unicode attack detected
Severity: Noise Reported: 68,258 times

These alerts are triggered by Unicode-encoded "\" or "/" characters on common HTTP ports and are typical behaviours associated with Code Red, Code RedII, Nimda, and Sadmind.

Fortunately, there is no such traffic to or from the inside the University's network. However, the fact that the IDS sensor(s) captured the alerts indicates that the either there is no filtering at the University's network border and/or the IDS sensor(s) is placed outside of the University network.

Noteworthy are the amount of alerts generated by two main IP addresses range, that of the University of Maryland and Deutsche Telekom (An ISP in Germany).

Correlations: Todd Beasley (GCIA Analyst #525) noted this event in his practical assignment. However, the data he analyzed indicated that the University's was infected with one of the worm.

Recommendations: Filter these traffics at the border or consider redeploying the IDS sensor(s). In additions, the University's representative should notify the owner(s) of the possibly infected machines.

spp_http_decode: CGI Null Byte attack detected
Severity: Noise Reported: 36,635 times

These alerts are triggered by the "%00" Null byte characters at the end of the CGI request and can cause the server which host the CGI scripts to leak proprietary information

Fortunately, there is no such traffic to or from the inside the University's network. However, the fact that the IDS sensor(s) captured the alerts indicates that the either there is no filtering at the University's network border and/or the IDS sensor(s) is placed outside of the University network.

Noteworthy are the amount of alerts generated by the University of Maryland.

Correlations: Michael Holstein (GCIA Analyst #529) noted this event in his practical assignment. However, the data he analyzed indicated that the University's was infected with one of the worm.

Recommendations: Filter these traffics at the border or consider redeploying the IDS sensor(s). In additions, the University's representative should notify the owner(s) of the possibly infected machines.

SMB Name Wildcard
Severity: Noise Reported: 26,387 times

These alerts are triggered by normal NetBIOS name resolution traffic.

Fortunately, there is no such traffic to or from the inside the University's network. However, the fact that the IDS sensor(s) captured the alerts indicates that the either there is no filtering at the University's network border and/or the IDS sensor(s) is placed outside of the University network.

Correlations: Michael Holstein (GCIA Analyst #529) noted this event in his practical assignment. The data he analyzed also indicated that the University's network did not leak NetBIOS traffic.

Recommendations: Filter these traffics at the border or consider redeploying the IDS sensor(s). In additions, the University's representative should notify the owner(s) of the possible misconfigured machines.

Watchlist 000220 IL-ISDNNET-990517
Severity: Noise Reported: 23,284 times

These alerts are triggered by traffic to and from ISDNNET (An ISP in Israel).

Noteworthy are the amount of traffic on port 80 from this ISP to the University of Maryland.

Correlations: Christopher Lee (GCIA Analyst #505) noted this event in his practical assignment. The data he analyzed also indicated that this did not affect the University's network.

Recommendations: Filter these traffics at the border or consider redeploying the IDS sensor(s).

TFTP - External UDP connection to internal tftp server
Severity: Noise Reported: 19,799 times

These alerts are triggered by normal TFTP traffic running utilizing UDP protocol and with what appears to be inbound connection.

Fortunately, there is no such traffic to or from the inside the University's network. However, the fact that the IDS sensor(s) captured the alerts indicates that the either there is no filtering at the University's network border and/or the IDS sensor(s) is placed outside of the University network.

Noteworthy are the amount of alerts generated by the University of Maryland and that Nimda uses TFTP to spread. If 192.168.0.216 is an internal University's IP address, then the verdict is at the opposite end of the spectrum.

Correlations: Michael Wilkinson (GCIA Analyst #508) noted this event in his practical assignment. He also indicated that this kind of traffic is highly suspicious.

Recommendations: Filter these traffics at the border or consider redeploying the IDS sensor(s). In additions, the University's representative should notify the owner(s) of the machines in question.

External RPC call
Severity: Noise Reported: 12,145 times

These alerts are triggered by normal RPC traffic on the External side.

Fortunately, there is no such traffic to or from the inside the University's network. However, the fact that the IDS sensor(s) captured the alerts indicates that the either there is no filtering at the University's network border and/or the IDS sensor(s) is placed outside of the University network.

```
| 8352     194.98.189.139->111 - External RPC call
|
| 2083     205.231.184.6->111 - External RPC call
|
| 917      203.239.155.2->111 - External RPC call
|
| 775      202.108.109.100->111 - External RPC call
|
| 11       66.32.232.141->111 - External RPC call
|
| 6        66.1.1.121->111 - External RPC call
|
| 1        203.239.155.2->41 - External RPC call
|
|
|
```

```
| Total Uniques:            7                    Total EOIs:    12145
|
```

Noteworthy are the amount of traffic to port 111 of machines residing at the University of Maryland. The majority of the connections are from Uunet France, Onramp (ISP in AL), Elimnet (ISP in Korea), and a Technology company in Beijing, China.

Correlations: Scott Shinberg (GCIA Analyst #389) noted this event in his practical assignment. The data he analyzed also indicated that the University's network was not affected.

Recommendations: Filter these traffics at the border or consider redeploying the IDS sensor(s). In additions, the University's representative should notify the owner(s) of the machines in question.

Alerts Concerning Trojan/Rootkit/Dangerous Activity

# of Alerts	Alert Message	Severity
3256	Possible trojan server activity	Noise
2483	SUNRPC highport access!	Noise
1658	IRC evil - running XDCC	Noise
490	High port 65535 udp - possible Red Worm - traffic	Noise
31	TFTP - Internal UDP connection to external tftp server	Noise
16	High port 65535 tcp - possible Red Worm - traffic	Noise
5	DDOS shaft client to handler	Noise
3	NIMDA - Attempt to execute cmd from campus host	Noise
2	tw33dl3	Noise
1	Back Orifice	Noise

These events are of interest due to their dangerous nature. Rootkits and Trojans are often used by attackers to compromise and retain control of a large number of servers. Fortunately, the University's network defense seems to be effective as no dangerous activity was observed from the internal network to external network or vice versa. As such, no further actions is required at this time other than to note that these alerts were raised by the Intrusion Detection System.

Alerts Top Talkers List

Sources

The table below lists the top 10 source IP addresses that are most active (raised the most number of alerts) during the period of July 30th through August 3rd. Overall, the top 10 sources account for 242,602 of 385,809 alerts (total by 11, 097 sources) or approximately 63% of the total.

IP Address	# of Alerts
63.250.213.12	109,417
3.0.0.99	42,230
130.85.81.37	27,094
130.85.85.74	19,122
80.145.95.201	15,205
194.98.189.139	8,375
212.179.35.118	6,213
130.85.111.230	5,016
63.250.213.73	4,975
130.85.111.231	4,955

Destination

The table below lists the top 10 destination IP addresses that are most active (raised the most number of alerts) during the period of July 30th through August 3rd. Overall, the top 10 destinations account for 169,864 of 385,809 alerts (total by 8,126 destinations) or approximately 44% of the total. Notice the oddity of the private IP address 10.0.0.1 and 192.168.0.216 as these are usually not routable in a public network.

IP Address	# of Alerts
233.28.65.148	109,410
10.0.0.1	42,230
216.241.219.28	30,877
233.2.171.1	25,919
192.168.0.216	19,793
207.200.86.97	10,140
207.200.86.66	9,503
233.28.65.173	4,975
130.85.104.204	3,911
130.85.154.27	3,106

Registration Information

Table below lists the registration information for the IP address that raised a number of suspicious alerts based on the seven alerts analyzed above. Most of the information was obtained from www.samspade.org or www.geektools.com unless otherwise noted.

IP Address	Domain Name Registration Information

63.250.213.12 63.250.213.73	OrgName: Yahoo! Broadcast Services, Inc. OrgID: YAHO NetRange: 63.250.192.0 - 63.250.223.255 CIDR: 63.250.192.0/19 NetName: NETBLK2-YAHOOBS NetHandle: NET-63-250-192-0-1 Parent: NET-63-0-0-0-0 NetType: Direct Allocation NameServer: NS1.YAHOO.COM NameServer: NS2.YAHOO.COM NameServer: NS3.YAHOO.COM NameServer: NS4.YAHOO.COM NameServer: NS5.YAHOO.COM Comment: ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE RegDate: 1999-11-24 Updated: 2002-03-27 TechHandle: NA258-ARIN TechName: Netblock Admin, Netblock TechPhone: +1-408-349-7183 TechEmail: netblockadmin@yahoo-inc.com
3.0.0.99	OrgName: General Electric Company OrgID: GENERA-9 NetRange: 3.0.0.0 - 3.255.255.255 CIDR: 3.0.0.0/8 NetName: GE-INTERNET NetHandle: NET-3-0-0-0-1 Parent: NetType: Direct Assignment Comment: RegDate: 1988-02-23 Updated: 1998-11-12 TechHandle: GET2-ORG-ARIN TechName: General Electric Company TechPhone: +1-518-612-6672 TechEmail: GENICTech@ge.com

130.85.81.37 130.85.85.74	OrgName: University of Maryland Baltimore County OrgID: UMBC NetRange: 130.85.0.0 - 130.85.255.255 CIDR: 130.85.0.0/16 NetName: UMBCNET NetHandle: NET-130-85-0-0-1 Parent: NET-130-0-0-0-0 NetType: Direct Assignment NameServer: UMBC5.UMBC.EDU NameServer: UMBC4.UMBC.EDU NameServer: UMBC3.UMBC.EDU Comment: RegDate: 1988-07-05 Updated: 2000-03-17 TechHandle: JJS41-ARIN TechName: Suess, John TechPhone: +1-410-455-2582 TechEmail: jack@umbc.edu

80.145.95.201	inetnum: 80.128.0.0 - 80.146.159.255 netname: DTAG-DIAL16 descr: Deutsche Telekom AG country: DE admin-c: DTIP-RIPE tech-c: ST5359-RIPE status: ASSIGNED PA remarks: ** *** remarks: * ABUSE CONTACT: abuse@t-ipnet.de IN CASE OF HACK ATTACKS, remarks: * ILLEGAL ACTIVITY, VIOLATION, SCANS, PROBES, SPAM, ETC. remarks: ** *** notify: auftrag@nic.telekom.de notify: dbd@nic.dtag.de mnt-by: DTAG-NIC changed: auftrag@nic.telekom.de 20020108 source: RIPE route: 80.128.0.0/11 descr: Deutsche Telekom AG, Internet service provider origin: AS3320 mnt-by: DTAG-RR changed: bp@nic.dtag.de 20010807 source: RIPE person: DTAG Global IP-Adressing address: Deutsche Telekom AG address: Bayreuther Strasse 1 address: D-90409 Nuernberg address: Germany phone: +49 911 68909856 e-mail: ripe.dtip@telekom.de nic-hdl: DTIP-RIPE mnt-by: DTAG-NIC changed: ripe.dtip@telekom.de 20020717 source: RIPE person: Security Team address: Deutsche Telekom AG address: Techniederlassung Schwaebisch Hall address: D-89070 Ulm address: Germany phone: +49 731 100 84055 fax-no: +49 731 100 84150 e-mail: abuse@t-ipnet.de nic-hdl: ST5359-RIPE notify: auftrag@nic.telekom.de notify: dbd@nic.dtag.de

194.98.189.139	inetnum: 194.98.189.128 - 194.98.189.143 netname: INGENCYS-NET1 descr: INGENCYS country: FR admin-c: DR5-RIPE tech-c: JB371-RIPE status: ASSIGNED PA remarks: abuse@fr.uu.net mnt-by: IWAY-NOC changed: frederic.martzel@mciworldcom.fr 20010924 source: RIPE route: 194.98.0.0/16 descr: UUNET-BLOCK1 descr: UUNET France Block 1 origin: AS702 remarks: *********************************** remarks: For all spamming or hacking problems remarks: please send your requests directly to remarks: abuse@fr.uu.net remarks: *********************************** notify: net-adm@mciworldcom.fr mnt-by: IWAY-NOC changed: net-adm@iway.fr 19981109 changed: frederic.martzel@mciworldcom.fr 20011114 source: RIPE role: technical contact address: UUNET FRANCE address: 215, Avenue Georges Clemenceau address: F-92024 NANTERRE Cedex phone: +33 1 56 38 22 00 fax-no: +33 1 56 38 22 01 e-mail: net-adm@mciworldcom.fr admin-c: VP1616-RIPE admin-c: FM7174-RIPE admin-c: AW7486-RIPE tech-c: ZM321-RIPE tech-c: AH6610-RIPE tech-c: TC334-RIPE nic-hdl: JB371-RIPE remarks: ------------------------------------ remarks: For all spamming or hacking problems remarks: please send your requests directly to remarks: abuse@fr.uu.net remarks: ------------------------------------ mnt-by: IWAY-NOC changed: frederic.martzel@mciworldcom.fr 20010828 source: RIPE

205.231.184.6	OrgName: UR*ONRAMP OrgID: URON NetRange: 205.231.184.0 - 205.231.191.255 CIDR: 205.231.184.0/21 NetName: ONRAMP-TUSCALOOSA-AL NetHandle: NET-205-231-184-0-1 Parent: NET-205-228-0-0-1 NetType: Reallocated Comment: RegDate: 1995-07-25 Updated: 1998-06-24 TechHandle: CW309-ARIN TechName: White, Craig TechPhone: +1-205-348-9690 TechEmail: cwhite@tusc.net

203.239.155.2 from whois.nic.or.kr	KRNIC is not ISP but National Internet Registry similar with APNIC. The IP address is allocated and still held by the following ISP, or they did not update whois information after assigning to end-user. Please see the following ISP contacts for relevant information or network abuse complaints. [ISP Organization Information] Org Name : ELIMNET, INC. Service Name : ELIMNET Org Address : 7F Choongjung Bldg, 32-11, Choongjungno 3-Ka Seodaemoon-Gu, Seoul, Korea [ISP IP Admin Contact Information] Name : YoungDae Seo Phone : +82-2-3149-4836 Fax : +82-2-3149-4998 E-Mail : nmc@elim.net [ISP IP Tech Contact Information] Name : JiYoung Hwang Phone : +82-2-3149-4835 Fax : +82-2-3149-4998 E-mail : domain@elim.net [ISP Network Abuse Contact Information] Name : JungHyun Noh Phone : +82-2-3149-4941 Fax : +82-2-3149-4998 E-mail : abuse@elim.net

202.108.109.100	inetnum: 202.108.109.0 - 202.108.109.255 netname: BJ-GX-DIGIT-TECH-CO descr: Beijing Guang Xinwang Digital descr: Technology Co.Ltd country: CN admin-c: HJ49-AP tech-c: HJ49-AP mnt-by: MAINT-CHINANET-BJ changed: suny@publicf.bta.net.cn 20020416 status: ALLOCATED PORTABLE source: APNIC person: He JianBo address: Dong Zhong Jie 9 Dong Cheng District address: Beijing 100027 phone: +86-10-64181150-215 fax-no: +86-10-64181819 e-mail: jumper@btamail.net.cn nic-hdl: HJ49-AP mnt-by: MAINT-CHINANET-BJ changed: suny@publicf.bta.net.cn 20000419 source: APNIC

233.28.65.173 233.28.65.148 233.2.171.1	OrgName: IANA OrgID: IANA-2 NetRange: 224.0.0.0 - 239.255.255.255 CIDR: 224.0.0.0/4 NetName: MCAST-NET NetHandle: NET-224-0-0-0-1 Parent: NetType: Direct Assignment NameServer: FLAG.EP.NET NameServer: STRUL.STUPI.SE NameServer: NS.ISI.EDU NameServer: NIC.NEAR.NET Comment: This block is reserved for special purposes. Please see RFC 3171 for additional information. RegDate: 1991-05-22 Updated: 2000-09-12 TechHandle: IANA-ARIN TechName: Internet Corporation for Assigned Names and Number TechPhone: +1-310-823-9358 TechEmail: res-ip@iana.org

Scans Data Analysis

A total of 2,671,215 events were captured and generated 13 different types of alerts between July 30th and August 3rd. The breakdown based on dates are as followed:

```
| 986120   Aug/3
|
| 622285   Aug/2
|
| 555154   Jul/30
|
| 331295   Jul/31
|
| 176361   Aug/1
|
```

The chart below breakdown the events by alert type:

| 2419257 UDP scan (Externally-based)
| 247710 SYN scan (Externally-based)
| 1163 NULL scan (Externally-based)

| 1072 NOACK scan (Externally-based)
| 893 INVALIDACK scan (Externally-based)
| 428 UNKNOWN scan (Externally-based)
| 384 VECNA scan (Externally-based)
| 58 SYNFIN scan (Externally-based)
| 57 FIN scan (Externally-based)
| 57 XMAS scan (Externally-based)
| 56 FULLXMAS scan (Externally-based)
| 48 NMAPID scan (Externally-based)
| 32 SPAU scan (Externally-based)

As seen in the table above, the majority of the events were raised by the first two alert types captured. These are possibly generated by Intrusion Detection System that are poorly configured and/or were not poorly deployed. Further exploration of these events is required to determine the possible cause.

Assumption: MY.NET is the prefix for the University's network.

Frequent Scan Details (Generated more than 10,000 events)

UDP Scan (Externally Based)
Severity: Medium Reported: 2,419,257 times

These UDP packets were captured by the Snort IDS sensor(s) did not appear to have originated from or destined to University's network. The majority of the traffic originated from the University of Maryland.

Almost 69% (1,657,985 of 2,419,257) of the UDP scan are being generated by 130.85.70.200 to 106,266 unique IP addresses. Almost all (1,657,765 of 1,657,985) of the scan from 130.85.70.200 are directed to port 41170. This port is associated with a Peer-2-Peer network called BLUSTER.

```
| 1657985  130.85.70.200
|
| 199851   130.85.70.207
|
| 171196   130.85.165.24
|
| 160532   130.85.82.2
|
| 62007    130.85.70.180
|
| 54892    130.85.137.7
|
| 31912    130.85.81.27
|
| 17469    130.85.87.44
|
```

```
| 12138    130.85.83.146
|
```

Correlations: Todd Beasley (GCIA Analyst #525) noted this event in his practical assignment. The data he analyzed also indicated that this did not affect the University's network and the threshold on IDS sensor(s) need to be adjusted.

Recommendations: Filter these traffics at the border or consider redeploying the IDS sensor(s). In addition, consider notifying the University of Maryland that their machine(s) might be participating in a Peer-2-Peer network.

SYN Scan (Externally Based)
Severity: Medium Reported: 247,710 times

These UDP packets were captured by the Snort IDS sensor(s) did not appear to have originated from or destined to University's network. The majority of the scans were directed at web, MS-SQL, FTP, Gnutella, SunRPC, and SMTP servers.

```
| 116765   80
|
| 60837    1433
|
| 28559    21
|
| 12030    6346
|
| 9873     111
|
| 5883     25
|
```

Correlations: Todd Beasley (GCIA Analyst #525) noted this event in his practical assignment. However, the data he analyzed was based on internal SYN scan caused by Windows logon sequence and tweaking of the IDS sensor(s) was recommended.

Recommendations: Filter these traffics at the border or consider redeploying the IDS sensor(s).

Alerts Concerning Unusual Scan

# of Alerts	Alert Message	Severity
1072	NOACK scan (Externally-based)	Noise
893	INVALIDACK scan (Externally-based)	Noise
428	UNKNOWN scan (Externally-based)	Noise
384	VECNA scan (Externally-based)	Noise
32	SPAU scan (Externally-based)	Noise

These events are of interest due to their unusual nature and are usually associated with stealth scanning techniques attempting to bypass firewall rules and IDS detection. Fortunately, none of these scan were directed at or from the University's network. As such, no further actions is required at this time other than to note that these alerts were raised by the Intrusion Detection System.

Scans Top Talkers List

Sources

The table below lists the top 10 source IP addresses that are most active (raised the most number of alerts) during the period of July 30th through August 3rd. Overall, the top 10 sources account for 2,386,508 of 2,671,215 alerts (total by 408 sources) or approximately 89% of the total. It should be noted that 2,375,769 of 2,671,215 alerts or approximately 89% of the total are generated by IP addresses that are associated with the University of Maryland.

IP Address	# of Alerts
130.85.70.200	1,659,446
130.85.70.207	199,851
130.85.165.24	171,223
130.85.82.2	160,603
130.85.70.180	62,220
130.85.137.7	60,876
130.85.81.27	31,926
130.85.87.44	17,470
130.85.83.146	12,154
202.98.223.86	10,739

Destination

The table below lists the top 10 destination IP addresses that are most active (raised the most number of alerts) during the period of July 30th through August 3rd. Overall, the top 10 destinations account for 67,505 of 2,671,215 alerts (total by 279,317 destinations) or approximately 3% of the total. Note that of the top 10 destinations, only the top destination has an abnormal amount of alerts and the remaining nine destinations are split evenly.

IP Address	# of Alerts
204.183.84.240	12,980
66.130.178.166	7,481
152.163.190.1	7,091
204.183.84.225	6,897
210.187.110.110	6,761

24.184.56.5	5,463
24.242.107.88	5,416
216.6.143.200	5,342
65.24.57.184	5,066
12.239.152.160	5,008

Registration Information

Table below lists the registration information for the IP address that raised a number of suspicious alerts based on the two alerts analyzed above and the top talkers list. Most of the information was obtained from www.samspade.org or www.geektools.com unless otherwise noted.

IP Address	Domain Name Registration Information
130.85.70.200 130.85.70.207 130.85.165.24 130.85.82.2 130.85.70.180 130.85.137.7 130.85.81.27 130.85.87.44 130.85.83.146	OrgName: University of Maryland Baltimore County OrgID: UMBC NetRange: 130.85.0.0 - 130.85.255.255 CIDR: 130.85.0.0/16 NetName: UMBCNET NetHandle: NET-130-85-0-0-1 Parent: NET-130-0-0-0-0 NetType: Direct Assignment NameServer: UMBC5.UMBC.EDU NameServer: UMBC4.UMBC.EDU NameServer: UMBC3.UMBC.EDU Comment: RegDate: 1988-07-05 Updated: 2000-03-17 TechHandle: JJS41-ARIN TechName: Suess, John TechPhone: +1-410-455-2582 TechEmail: jack@umbc.edu

202.98.223.86	inetnum: 202.98.192.0 - 202.98.223.255 netname: CHINANET-GZ descr: CHINANET Guizhou province network descr: Data Communication Division descr: China Telecom country: CN admin-c: CH93-AP tech-c: DL72-AP mnt-by: MAINT-CHINANET mnt-lower: MAINT-CHINANET-GUIZHOU changed: hostmaster@ns.chinanet.cn.net 20000101 status: ALLOCATED PORTABLE source: APNIC person: Chinanet Hostmaster address: No.31 ,jingrong street,beijing address: 100032 country: CN phone: +86-10-66027112 fax-no: +86-10-66027334 e-mail: hostmaster@ns.chinanet.cn.net nic-hdl: CH93-AP mnt-by: MAINT-CHINANET changed: hostmaster@ns.chinanet.cn.net 20020814 source: APNIC person: dan lu address: 93.south zhonghua road of guiyang address: 550001 china country: CN phone: +86-851-6861469 fax-no: +86-851-6861469 e-mail: ljt@public.gz.cn nic-hdl: DL72-AP mnt-by: MAINT-CHINANET-GUIZHOU changed: ljt@public.gz.cn 20001218 changed: ljt@public.gz.cn 20020402 source: APNIC

204.183.84.24 0	OrgName: Ashby & Geddes OrgID: ASHBYG NetRange: 204.183.84.0 - 204.183.84.255 CIDR: 204.183.84.0/24 NetName: ASGE001-204-183-84 NetHandle: NET-204-183-84-0-1 Parent: NET-204-183-80-0-1 NetType: Reassigned Comment: RegDate: 1998-09-30 Updated: 1998-09-30 TechHandle: AG89-ARIN TechName: Geddes, Ashby TechPhone: +1-302-654-1888 TechEmail: dns@dca.net

Out of Spec (OOS) Data Analysis

A total of 1,636 Out of Specification packets were captured between July 31st and August 4th. The breakdown based on dates are as followed:

```
| 1124      Aug/01
|
| 503       Aug/02
|
| 7         Jul/31
|
| 1         Aug/04
|
| 1         Aug/03
|
```

The SNORT IDS sensor(s) triggered the alerts for OOS packets due to the following reasons:

- Packet Corruption
- Implementation of Explicit Congestion Notification standard (RFC2481)
- Crafted packets
- Packets that do not match a particular Snort alert rule.

Implementations of the TCP/IP stack on some Operating System are not robust enough to handle exceptions. As a result, they will re-act in unexpected manners when they received these OOS packets. This usually resulted in a system crash, information leakage, or possibly system compromise.

All 1,636 were of external origins from 74 unique IP addresses and were destined for 45 unique internal IP addresses belonging to the University. The top 3 sources of IP addresses sending out more than 100 OOS packets are listed below:

IP Address	# of Alerts	Ports
68.32.126.64	652	110
62.76.241.129	345	113
209.116.70.75	214	25

Looking at the sample probe from 68.32.126.64:

```
08/01-01:15:00.857663 68.32.126.64:26163 -> MY.NET.6.7:110
TCP TTL:48 TOS:0x0 ID:2006  DF
21S***** Seq: 0x12602526   Ack: 0x0   Win: 0x16D0
TCP Options => MSS: 1460 SackOK TS: 52306672 0 EOL EOL EOL EOL
```

And for 62.76.241.129:

```
08/01-01:18:33.942643 62.76.241.129:38365 -> MY.NET.97.217:113
TCP TTL:45 TOS:0x0 ID:13007  DF
21S***** Seq: 0x2E003F94   Ack: 0x0   Win: 0x16D0
TCP Options => MSS: 1460 SackOK TS: 59969158 0 EOL EOL EOL EOL
```

And for 209.116.70.75:

```
08/01-01:28:43.684212 209.116.70.75:41637 -> MY.NET.100.217:25
TCP TTL:51 TOS:0x0 ID:42455  DF
21S***** Seq: 0x53B7D9D3   Ack: 0x0   Win: 0x16D0
TCP Options => MSS: 1460 SackOK TS: 770604036 0 EOL EOL EOL EOL
```

The flags "21S*****" indicates that the initial SYN is performing a network congestion check as per ECN standard. Further analysis via grep revealed that of the 1636 OOS packets, 1616 are in this category.

Based on their fingerprint, there is a chance that these packets came from platforms that are based on Linux 2.2.x or OS/400 R4.4 or Solaris 8 OS.

Noteworthy are the number of what appear to be crafted packets from machines under this category:

Suspicious IP address 61.170.132.27

```
08/01-02:48:18.258649 61.170.132.27:1363 -> MY.NET.111.140:103
TCP TTL:46 TOS:0x0 ID:57131  DF
```

```
21*FR*** Seq: 0x500013   Ack: 0x318C9256   Win: 0xA010
TCP Options => EOL EOL NOP NOP
C2 D1                                            ..
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-02:48:20.657415 61.170.132.27:1365 -> MY.NET.111.140:80
TCP TTL:46 TOS:0x0 ID:17452  DF
*1SF*P*U Seq: 0x13318F   Ack: 0x91D70001   Win: 0x5010
00 00 00 00 00 00                                ......
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-02:48:50.620950 61.170.132.27:1355 -> MY.NET.111.140:80
TCP TTL:46 TOS:0x0 ID:30512  DF
21*F**** Seq: 0x1332AC   Ack: 0x92690000   Win: 0x5010
TCP Options => EOL EOL EOL EOL EOL EOL SackOK EOL
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-02:49:01.364194 61.170.132.27:1361 -> MY.NET.111.140:80
TCP TTL:46 TOS:0x0 ID:64561  DF
2*SF**** Seq: 0x1332B4   Ack: 0x91990001   Win: 0x5010
TCP Options => EOL EOL EOL EOL EOL EOL SackOK
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-02:49:05.227124 61.170.132.27:1356 -> MY.NET.111.140:80
TCP TTL:46 TOS:0x0 ID:37682  DF
21SFR*A* Seq: 0x1332AD   Ack: 0x92090000   Win: 0x5010
00 00 00 00 00 00                                ......
```

Based on the above information, there is a chance that these packets came from a platform that is based on Cisco IOS 11.2.

Suspicious IP address 61.151.232.174

```
08/01-04:13:25.436547 61.151.232.174:1610 -> MY.NET.111.140:0
TCP TTL:46 TOS:0x0 ID:28849  DF
2*SFR**U Seq: 0x500041   Ack: 0xE0ADD404   Win: 0x5010
TCP Options => EOL EOL EOL EOL
00 00                                            ..
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-04:13:50.105369 61.151.232.174:1616 -> MY.NET.111.140:80
TCP TTL:46 TOS:0x0 ID:39860  DF
21SFRPAU Seq: 0x41E206   Ack: 0xD4BE000D   Win: 0x5010
TCP Options => EOL EOL EOL EOL
00 00                                            ..
```

Based on the above information, there is a chance that these packets came from a platform that is based on Cisco IOS 11.2 or Linux 2.2.x.

IP address 211.154.85.159

```
08/01-04:21:15.877198 211.154.85.159:1722 -> MY.NET.111.140:80
TCP TTL:107 TOS:0x0 ID:21989  DF
21S*R*AU Seq: 0x1A7   Ack: 0xC35BF035   Win: 0x5010
TCP Options => EOL EOL EOL EOL EOL EOL
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-05:19:16.649660 211.154.85.159:1754 -> MY.NET.111.140:80
TCP TTL:107 TOS:0x0 ID:1658  DF
21S***AU Seq: 0x1DB6FB5   Ack: 0xC5A0B5B1   Win: 0x5010
06 DA 00 50 01 DB 6F B5 C5 A0 B5 B1 00 F2 50 10  ...P..o.......P.
```

```
B5 80 EB 2B 00 00 00 00 00 00                       ...+......
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-05:19:16.920123 211.154.85.159:1753 -> MY.NET.111.140:80
TCP TTL:107 TOS:0x0 ID:3194
**SF***U Seq: 0x1DB1677   Ack: 0xF0C438   Win: 0x5010
TCP Options => EOL EOL EOL EOL EOL EOL SackOK
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-05:34:08.929013 211.154.85.159:1787 -> MY.NET.111.140:80
TCP TTL:107 TOS:0x0 ID:33153  DF
2*SF**** Seq: 0x1EA57B7   Ack: 0x3915778   Win: 0x5010
06 FB 00 50 01 EA 57 B7 03 91 57 78 00 43 50 10  ...P..W...Wx.CP.
B4 CD 23 F6 00 00 00 00 00 00                       ..#.......
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-05:38:52.122492 211.154.85.159:1798 -> MY.NET.111.140:80
TCP TTL:107 TOS:0x0 ID:53641  DF
2*SFR*A* Seq: 0x1EE7F29   Ack: 0x149BC537   Win: 0x5010
07 06 00 50 01 EE 7F 29 14 9B C5 37 00 57 50 10  ...P...)...7.WP.
B0 8C 81 EC 00 00 00 00 00 00                       ..........
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-05:46:41.855061 211.154.85.159:0 -> MY.NET.111.140:1816
TCP TTL:107 TOS:0x0 ID:24217  DF
*1SF**AU Seq: 0x5001F5   Ack: 0x344D3196   Win: 0x5010
00 00 07 18 00 50 01 F5 34 4D 31 96 09 B3 50 10  .....P..4M1...P.
B5 80 66 45 00 00 00 00 00 00                       ..fE......
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-05:57:29.626865 211.154.85.159:20 -> MY.NET.111.140:1852
TCP TTL:107 TOS:0x0 ID:44972
21S***A* Seq: 0x5001FE   Ack: 0xE56958A0   Win: 0x5010
TCP Options => EOL EOL EOL EOL EOL EOL WS: 1 NOP TS: 3604480 0 EOL
EOL EOL EOL EOL EOL EOL EOL EOL EOL EOL EOL EOL EOL EOL EOL
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-06:13:00.731738 211.154.85.159:1893 -> MY.NET.111.140:80
TCP TTL:107 TOS:0x0 ID:59605  DF
21S*R*** Seq: 0x20DB060   Ack: 0x94F7   Win: 0x5010
TCP Options => EOL EOL EOL EOL EOL EOL
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-06:42:01.892932 211.154.85.159:1959 -> MY.NET.111.140:80
TCP TTL:107 TOS:0x0 ID:10526  DF
*1SF**A* Seq: 0x2281695   Ack: 0x2ADA817   Win: 0x5010
07 A7 00 50 02 28 16 95 02 AD A8 17 00 93 50 10  ...P.(........P.
B5 80 13 C0 00 00 00 00 00 00                       ..........
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-06:49:03.188702 211.154.85.159:1975 -> MY.NET.111.140:80
TCP TTL:107 TOS:0x0 ID:46631  DF
21S**P** Seq: 0x22DC827   Ack: 0x1B1F351F   Win: 0x5010
07 B7 00 50 02 2D C8 27 1B 1F 35 1F 00 CA 50 10  ...P.-.'..5...P.
B5 80 BC 9E 00 00 00 00 00 00                       ..........
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-03:20:12.727003 211.154.85.159:0 -> MY.NET.111.140:1663
TCP TTL:107 TOS:0x0 ID:9575  DF
21S**PAU Seq: 0x500170   Ack: 0x24010A75   Win: 0x8010
TCP Options => EOL EOL NOP NOP
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-03:34:42.575935 211.154.85.159:0 -> MY.NET.111.140:1676
TCP TTL:107 TOS:0x0 ID:56950  DF
2*SF**A* Seq: 0x50017D   Ack: 0x95F442CC   Win: 0x8010
3C 53 80 10 B5 80 32 B5 00 00 01 01 05 0A 42 CC  <S....2.......B.
4D 57 42 CC                                         MWB.
```

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-03:41:31.177676 211.154.85.159:1681 -> MY.NET.111.140:80
TCP TTL:107 TOS:0x0 ID:62599  DF
21S***A* Seq: 0x18388F1   Ack: 0x145B3C   Win: 0x5010
TCP Options => EOL EOL EOL EOL EOL EOL SackOK
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-03:41:52.984402 211.154.85.159:182 -> MY.NET.111.140:1681
TCP TTL:107 TOS:0x0 ID:45962  DF
2*SFRPA* Seq: 0x500183   Ack: 0x8E9B5B48   Win: 0x5010
00 B6 06 91 00 50 01 83 8E 9B 5B 48 07 5F 50 10  .....P....[H._P.
B5 80 E5 91 00 00 00 00 00 00                    ..........
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-03:46:53.173239 211.154.85.159:1684 -> MY.NET.111.140:80
TCP TTL:107 TOS:0x0 ID:47505  DF
*1SFR*** Seq: 0x10187   Ack: 0xDE6E6D93   Win: 0x5010
TCP Options => EOL EOL EOL EOL EOL EOL
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-03:47:11.134688 211.154.85.159:0 -> MY.NET.111.140:1685
TCP TTL:107 TOS:0x0 ID:44435  DF
21*FRPAU Seq: 0x500188   Ack: 0x19436DB3   Win: 0x5010
B2 AD 08 AB 00 00 00 00 00 00                    ..........
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-04:03:22.663985 211.154.85.159:1694 -> MY.NET.111.140:80
TCP TTL:107 TOS:0x0 ID:55472  DF
**SFRPAU Seq: 0x197   Ack: 0xF031AEE7   Win: 0x5010
TCP Options => EOL EOL EOL EOL EOL EOL SackOK NOP NOP
```

Based on the above information, there is a chance that these packets came from a platform that is based on Netware 4.11 or Windows 2000/XP.

Notice the port 0 and the illegal flags on some of the packets. It seems that MY.NET.111.140 is targeted by the 3 IP addresses above. It is recommended that this machine be inspected for sign of impact.

IP address 68.52.37.114

```
08/02-18:18:03.509676 68.52.37.114:1682 -> MY.NET.163.107:6347
TCP TTL:111 TOS:0x0 ID:16477  DF
21*FR*** Seq: 0x104ADA2   Ack: 0x2A5D063D   Win: 0x8010
06 92 18 CB 01 04 AD A2 2A 5D 06 3D 00 C5 80 10  ........*].=....
F3 FF 88 D0 00 00 01 01 05 0A 2A 5D 0B F1 2A 5D  ..........*]..*]
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/02-18:18:37.543334 68.52.37.114:1682 -> MY.NET.163.107:6347
TCP TTL:111 TOS:0x0 ID:20588  DF
21SFR**U Seq: 0x104ADA2   Ack: 0x2A8C723D   Win: 0x5010
06 92 18 CB 01 04 AD A2 2A 8C 72 3D 00 E7 50 10  ........*.r=..P.
FF FF B5 A0 00 00 00 00 00 00                    ..........
```

Based on the above information, there is a chance that these packets came from a platform that is based on Netware 4.11 or Windows 2000/XP.

This IP address is searching for Gnutella on MY.NET.163.107.

The remaining packets are:

```
07/31-03:01:26.781451 4.64.202.110:22690 -> MY.NET.88.162:1607
TCP TTL:114 TOS:0x0 ID:64352  DF
**SFR*A* Seq: 0x4BEED5A   Ack: 0x57781947   Win: 0x5010
TCP Options => EOL EOL EOL EOL EOL EOL
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
07/31-13:47:46.826996 217.81.180.174:1699 -> MY.NET.150.225:1214
TCP TTL:116 TOS:0x0 ID:32639  DF
21**RPAU Seq: 0x610000   Ack: 0xBD42331F   Win: 0x5010
BD 42 33 1F 2B FC 50 10 7F FF 60 7E 00 00 00 00  .B3.+.P...`~....
00 00                                            ..
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/01-08:15:56.561821 12.217.148.206:4667 -> MY.NET.80.143:6375
TCP TTL:114 TOS:0x0 ID:48325  DF
21*FRPAU Seq: 0x1B6A237   Ack: 0x5522C0CF   Win: 0x5018
12 3B 18 E7 01 B6 A2 37 55 22 C0 CF 00 FD 50 18  .;.....7U"....P.
70 25 04 4A 00 00 9D 12 E3 E0 99 5E 3F 0C 55 6A  p%.J.......^?.Uj
5B F8                                            [.
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
08/02-15:52:30.752770 142.173.193.40:6346 -> MY.NET.153.160:2987
TCP TTL:113 TOS:0x0 ID:24922  DF
2*SFR**U Seq: 0x8FB   Ack: 0xF49D008C   Win: 0x5018
TCP Options => EOL EOL Opt 214 NOP Opt 69 (20): 5229 BAB4 1523 8908
0000 0000 0000 0000 0000 EOL EOL EOL EOL EOL EOL EOL EOL EOL EOL EOL
EOL
```

Based on the above information, there is a chance that these packets came from a platform that is based on Netware 4.11 or Windows 2000/XP.

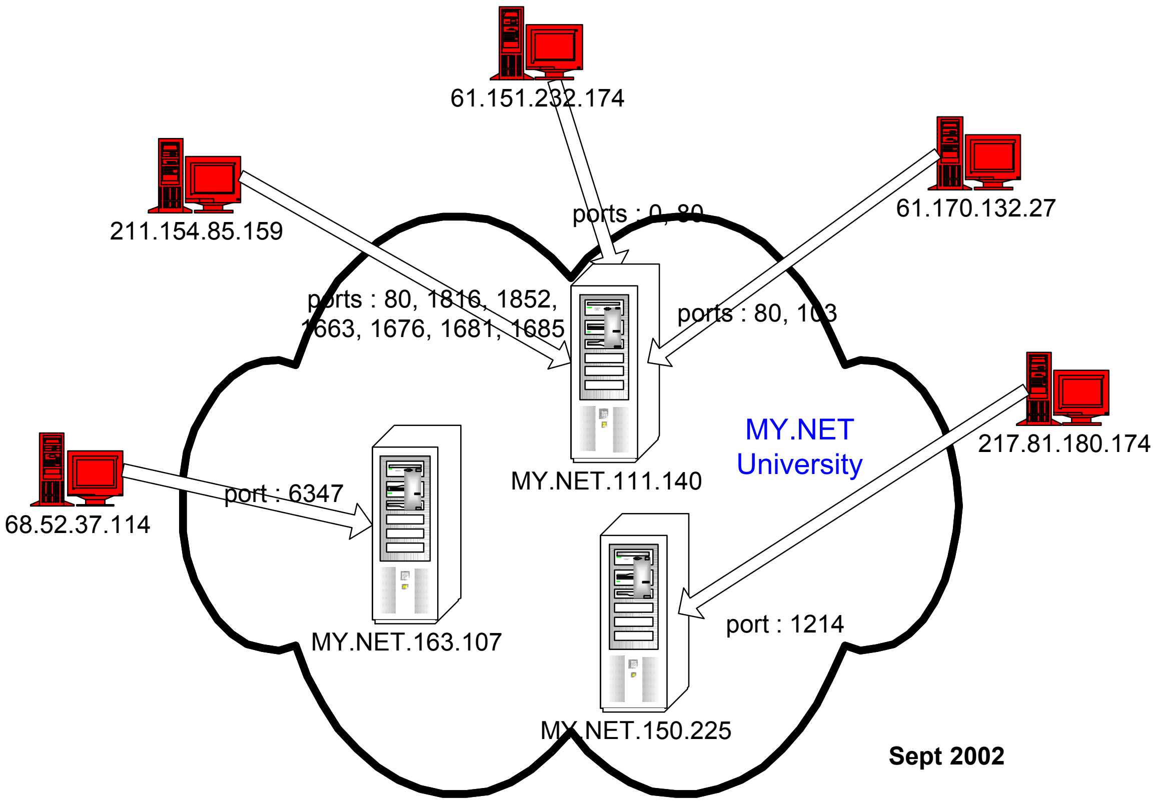
The remaining 4 are attributed to:

4.64.202.110 – scanning for SST service? or random scan. Log and no further action
217.81.180.174 – scanning for KAZAA at MY.NET.150.225. Log and no further action
12.217.148.206 – appears to be a random scan. Log and no further action
142.173.193.40 – scanning for identify service? or random scan. Log and no further action

There was no alert raised from Internal machines to External machines. This is a good thing as this means that University's network is well protected. Although, scans are happening, there is no impact at this time.

It should be noted that lots of these crafted packets are originating from China and these sources should be put on a "watch" list for potentially dangerous activities.

Correlations: Todd Beasley (GCIA Analyst #525) noted the ECN events in his practical assignment. The data he analyzed also indicated that this did not affect the University's network.



MY.NET Suspicious OOS Link Graph

Recommendations: Check MY.NET.111.140 for impact by OOS packets.
Check MY.NET.163.107 for Gnutella software
Check MY.NET.150.225 for KAZAA software

OOS Top Talkers List

Sources

The table below lists the top 10 source IP addresses that are most active (raised the most number of alerts) during the period of July 31st through August 4th. Overall, the top 10 sources account for 1,442 of 1,636 alerts (total by 74 sources) or approximately 88% of the total.

IP Address	# of Alerts
68.32.126.64	652
62.76.241.129	345
209.116.70.75	214
212.35.180.17	83
65.210.154.210	48
213.250.44.19	29
202.155.91.142	18

61.132.74.239	18
209.132.232.101	18
211.154.85.159	17

Destination

The table below lists the top 10 destination IP addresses that are most active (raised the most number of alerts) during the period of July 31st through August 4th. Overall, the top 10 destinations account for 1,394 of 1,636 alerts (total by 45 destinations) or approximately 85% of the total.

IP Address	# of Alerts
MY.NET.6.7	660
MY.NET.97.217	241
MY.NET.97.238	104
MY.NET.100.217	95
MY.NET.253.20	85
MY.NET.111.198	54
MY.NET.100.165	43
MY.NET.253.125	41
MY.NET.253.114	37
MY.NET.6.40	34

Registration Information

Table below lists the registration information for the IP address that raised a number of suspicious alerts based on the alerts analyzed above. Most of the information was obtained from www.samspade.org or www.geektools.com unless otherwise noted.

IP Address	Domain Name Registration Information

61.170.132.27	inetnum: 61.169.0.0 - 61.171.255.255 netname: CHINANET-SH descr: CHINANET Shanghai province network descr: Data Communication Division descr: China Telecom country: CN admin-c: CH93-AP tech-c: XI5-AP mnt-by: MAINT-CHINANET mnt-lower: MAINT-CHINANET-SH changed: hostmaster@ns.chinanet.cn.net 20001201 status: ALLOCATED PORTABLE source: APNIC person: Chinanet Hostmaster address: No.31 ,jingrong street,beijing address: 100032 country: CN phone: +86-10-66027112 fax-no: +86-10-66027334 e-mail: hostmaster@ns.chinanet.cn.net nic-hdl: CH93-AP mnt-by: MAINT-CHINANET changed: hostmaster@ns.chinanet.cn.net 20020814 source: APNIC person: Wu Xiao Li address: Room 805,61 North Si Chuan Road,Shanghai,200085,PRC country: CN phone: +86-21-63630562 fax-no: +86-21-63630566 e-mail: ip-admin@mail.online.sh.cn nic-hdl: XI5-AP mnt-by: MAINT-CHINANET-SH changed: ip-admin@mail.online.sh.cn 20010510 source: APNIC

61.151.232.174	inetnum: 61.151.0.0 - 61.151.255.255 netname: CHINANET-SH descr: CHINANET Shanghai province network descr: Data Communication Division descr: China Telecom country: CN admin-c: CH93-AP tech-c: XI5-AP mnt-by: MAINT-CHINANET mnt-lower: MAINT-CHINANET-SH changed: hostmaster@ns.chinanet.cn.net 20000701 status: ALLOCATED PORTABLE source: APNIC person: Chinanet Hostmaster address: No.31 ,jingrong street,beijing address: 100032 country: CN phone: +86-10-66027112 fax-no: +86-10-66027334 e-mail: hostmaster@ns.chinanet.cn.net nic-hdl: CH93-AP mnt-by: MAINT-CHINANET changed: hostmaster@ns.chinanet.cn.net 20020814 source: APNIC person: Wu Xiao Li address: Room 805,61 North Si Chuan Road,Shanghai,200085,PRC country: CN phone: +86-21-63630562 fax-no: +86-21-63630566 e-mail: ip-admin@mail.online.sh.cn nic-hdl: XI5-AP mnt-by: MAINT-CHINANET-SH changed: ip-admin@mail.online.sh.cn 20010510 source: APNIC

211.154.85.159	inetnum: 211.154.85.1 - 211.154.85.255 netname: XUHUI2POPNET descr: Cable OnLine Network Xuhui2 pop. descr: Internet Service Provider descr: Shanghai China country: CN admin-c: HL6-CN tech-c: YM2-CN mnt-by: MAINT-CNNIC-AP changed: leion@cableplus.com.cn 20010615 status: ASSIGNED NON-PORTABLE source: APNIC changed: hm-changed@apnic.net 20020827 person: Huaiyu Li address: Computer Center address: Shanghai Cable TV Station address: 487#, East Luo Chuan Road, Shanghai 200072, China country: CN phone: +86 21 56729282 e-mail: fyma@shnet.edu.cn nic-hdl: HL6-CN mnt-by: MAINT-CN-CJJ changed: cjj@cableplus.com.cn 20010609 source: APNIC person: Yougang Min address: Computer Center address: Shanghai Cable TV Station address: 487#, East Luo Chuan Road, Shanghai 200072, China phone: +86 21 56729282 e-mail: fyma@shnet.edu.cn nic-hdl: YM2-CN mnt-by: MAINT-CN-CJJ changed: cjj@cableplus.com.cn 20010611 source: APNIC

68.32.126.64	CustName: Comcast Cable Communications, Inc. Address: 3 Executive Campus Cherry Hill, NJ 08002 Country: US Comment: RegDate: 2002-06-15 Updated: 2002-06-15 NetRange: 68.32.112.0 - 68.32.143.255 CIDR: 68.32.112.0/20, 68.32.128.0/20 NetName: JUMPSTART-BALTIMOR-A3 NetHandle: NET-68-32-112-0-1 Parent: NET-68-32-0-0-1 NetType: Reassigned Comment: RegDate: 2002-06-15 Updated: 2002-06-15

62.76.241.129	inetnum: 62.76.240.0 - 62.76.243.255 netname: UDMEDU-NET descr: Internet Center of Udmurt State University descr: ul. Universitetskaja, 1, k.6, Izhevsk, Russia country: RU admin-c: BGA4-RIPE tech-c: DMIR-RIPE status: ASSIGNED PA notify: dm@uni.udm.ru mnt-by: ROSNIIROS-MNT changed: ip-dbm@ripn.net 20000128 source: RIPE route: 62.76.240.0/22 descr: UDMEDU-NET origin: AS13094 mnt-by: UDSU-MNT changed: dm@uni.udm.ru 20010124 source: RIPE person: Basil G. Ananin address: ul. Universitetskaja, 1, k.6, room 320 address: Internet Center of UdSU address: Izhevsk, Russia phone: +7 3412788697 fax-no: +7 3412788697 e-mail: anan@uni.udm.ru nic-hdl: BGA4-RIPE notify: dm@uni.udm.ru notify: ip-reg@ripn.net changed: dm@uni.udm.ru 20000128 source: RIPE person: Dmitry N. Mironov address: 1, ul. Universitetskaja address: Izhevsk address: Russia phone: +7 3412 751758 fax-no: +7 3412 788697 e-mail: ddd@uni.udm.ru nic-hdl: DMIR-RIPE notify: ddd@uni.udm.ru changed: ddd@uni.udm.ru 19981020 source: RIPE

209.116.70.75	OrgName: Red Hat Inc. OrgID: REDHAT-1 NetRange: 209.116.70.64 - 209.116.70.95 CIDR: 209.116.70.64/27 NetName: INFLOW-RHAT1 NetHandle: NET-209-116-70-64-1 Parent: NET-209-116-68-0-1 NetType: Reassigned Comment: RegDate: 2001-02-02 Updated: 2001-08-23 TechHandle: AC812-ARIN TechName: Abuse Coordinator, Abuse TechPhone: +1-919-287-1100 TechEmail: abuse@inflow.com

Conclusion and Defensive Recommendations

Thorough analysis of the supplied logs indicated that the University has a good security measure in place. However, overall security posture can be enhanced via the following recommendations:

- Validate that ingress / egress filtering is implemented at the border routers and/or firewalls.
- Revisit the current sensor(s) deployment as to reduce the number of false positives. Currently the overwhelming number of alerts speaks volume against the return on investment of the existing Intrusion Detection System.
- Be proactive against security breaches by conducting regular vulnerability assessments of the University's network for known and new vulnerabilities.
- Implement a tracking mechanism to keep track of repeat offenders over an extended timeframe. This is also useful to detect the slow and meticulous attackers.
- Contact University of Maryland and inform them of the abundant activities (scans and alerts) associated with their IP addresses.
- Ensure that MY.NET.111.140 is not impacted by OOS packets.

- Check MY.NET.163.107 for Gnutella software

- Check MY.NET.150.225 for KAZAA software

Analysis Process

Major platforms, tools and services used in the analysis include:

- Slackware 8.1 and unix wonderful tools (on an IBM T23 1.13GHz 512MB)
- Microsoft Windows 2000 (on an IBM T23 1.13GHz 512MB)
- Microsoft Word 2000
- Microsoft Excel 2000
- ActiveState ActivePerl, Build 633(Perl v5.6.1 built for MSWin32-x86-multi-thread)
- csv.pl and summarize.pl script by Todd Beasley (GCIA Analyst #525)
- Snort 1.8.7 (rulesets and source code)
- Google (http://www.google.com) (Easily, the most used research tool)
- Sam Spade (http://www.samspade.org)
- Geektools (http://www.geektools.com)
- The SANS Institute (http://www.sans.org)

The original data from Alerts files for 5 days were concatenated into a single file before analysis. In additions, spp_portscan data were removed as these events are also found in the Scans data files. Similar consolidation process was completed for the Scans file and the OOS files. Pre-inspections of the data files revealed that only the OOS files have any MY.NET. annotation for the University's IP addresses. Some of the data in Alerts and OOS files needed to be clean up as some dates were incomplete and some of the lines were missing return or newline characters. However, not all the data could be cleanup in the Alerts file and missing data were replaced with a token string.

Attempts to use SnortSnarf on the Alerts file was futile and was a waste of time since SnortSnarf would crash after running for almost 1 full day. At this point, I decided to borrow the perl scripts written by Todd Beasley and with some minor modifications, was able to process the data files in record times. Unfortunately, the csv.pl scripts provided by Todd were written only for the Alerts and Scans files. Minor modifications to the csv.pl script were required to process the OOS files. Analysis of the data files posses little challenge with these scripts and grep. However, there were instances where system resources were exceeded and the workload were simplified in order to resolve the issue.

References

Beasley, Todd. "GCIA Practical Assignment." GIAC Certified Intrusion Analysts (GCIA #525). May 31, 2002. URL: http://www.giac.org/practical/Tod_Beardsley_GCIA.doc

(Sept 4, 2002)

Holdstein, Michael. "GCIA Practical Assignment." GIAC Certified Intrusion Analysts (GCIA #529). June 30, 2002. URL: http://www.giac.org/practical/Michael_Holstein_GCIA.doc (Sept 4, 2002)

Kuethe, Chris. "GCIA Practical Assignment." GIAC Certified Intrusion Analysts (GCIA #303). Feb 22, 2001. URL: http://www.giac.org/practical/chris_kuethe_gcia.html (Sept 4, 2002).

Lee, Christopher. "GCIA Practical Assignment." GIAC Certified Intrusion Analysts (GCIA #505). April 8, 2002. URL: http://www.giac.org/practical/Christopher_Lee_GCIA.doc (Sept 4, 2002)

Miller, Toby. "ECN and its Impact on Intrusion Detection". URL: http://www.incidents.org/detect/ecn.html (Sept 4, 2002)

Shinberg, Scott. "GCIA Practical Assignment." GIAC Certified Intrusion Analysts (GCIA #389). July 27, 2001. URL: http://www.giac.org/practical/Scott_Shinberg_GCIA.doc (Sept 4, 2002)

Spitzner, Lance. "Lists of fingerprints for passive fingerprint monitoring" May 23, 2000. URL: http://project.honeynet.org/papers/finger/traces.txt (Sept 4, 2002).

Wilkinson, Michael. "GCIA Practical Assignment." GIAC Certified Intrusion Analysts (GCIA #508). April 9, 2002. URL: http://www.giac.org/practical/michael_wilkinson_gcia.doc (Sept 4, 2002)