GIAC Certified Intrusion
Analyst (GCIA)
Practical Assignment
Version 3.5
Authored By: Ryan Ettl
<10-31-2004>

[Option #1 - Enterprise
IDS Architecture]

<Ryan Ettl>
<Track 3 Mentoring
Program>

Table of Contents

Deleted: → - → -

Deleted: - -

Deleted: → - → -

Deleted: - -

Abstract

Part I describes two separate methods of implementing an enterprise Cisco Network Intrusion Detection System. The first network traffic collection infrastructure describes a spanned port methodology, while the second describes a tapped infrastructure. Both I DS insertion methods will forward alerts to an ArcSight three tiered architecture for data consolidation.

Part II analyzes two detects taken from the enterprise Network intrusion detection system explained in part I. The first detect is pulled from http://www.incidents.org/ , while the second two detects are pulled from the architecture described in Part I.

Part III is the analysis of university logs analyzing the Top 20 source and Destination hosts.

Document Con ventions

`Command`	Operating system commands are represented in this font style. This style indicates a command that is entered at a command prompt or shell.
`computer output`	The results of a command and other computer output are in this style

Definitions:
Tier I = Facility with 5,000 employees or more
Tier II = Facility with 1000 employees or more, but less than 5,000
Tier III = Facility with 100 employees or less
SAN = Storage Area Network
Wd = WinDump

Deleted: → - → -

Deleted: - -

Part One: Design of an Enterprise IDS Architecture

Why an IDS solution? Large organizations today are implementing IDS solutions into their existing IT infrastructure for one or more of the summarized reasons below:

1. Help achieve Business Continuity
2. Protect the Confidentiality of Data from unauthorized disclosure
3. Help support high Availability of computing resources
4. Protect the Integrity of data from unauthorized modification [1]

Executive Summary of the Network

The first section of part I centers on the characteristics of the Cisco IDS 4235 and 4250 models, then shifts focus on IDS placement within an existing corporate infrastructure while addressing challenges of network data collection. A great deal of attention has been focused on how to create an IDS solution for a corporate DMZ architecture. However, “ Track 3 – Intrusion Detection in Depth” curriculum does not address IDS implementations or solutions for internal WAN infrastructure. This paper will utilize two network diagrams to describe where and how to insert Intrusion Detection Systems into an existing corporate WAN infrastructure using spanned and tapped methodologies . Throughout this section of the paper there will be references to Tier I, II and III facilities. The first network Diagram is referenced as Tier I. Tier I is defined as a large facility (greater than 5,000 employees) with centrally located corporate data services . The second diagram is referenced as Tier II. Tier II is defined as a medium size facility (greater than 1000 employees) with distributed corporate data services. Both Tier I and II have many isolated LAN subnets. Network traffic flows and data collection from these isolated subnets will be described in detail , including challenges with each architecture . Lastly and for reference, Tier III sites (100 or less employees) all access critical information across the Tier I and II internal WAN infrastructure. All facilities intercommunicate.

The second section of part I addresses the alert handling architecture. This section focuses on what to do with the network traffic once it has been collected and processed by the Cisco NIDS. An alert flow diagram will explain how alerts get to a centralized monitoring console using the ArcSight infrastructure , then moves on to describe 24x7 alert monitoring operations, procedures, and legal issues.

IDS System Description

There are two IDS models shown in the below network architecture diagrams: the Cisco NIDS 4235 and 4250 . The 4235 model is installed with the tapped

[1] Trinity Security Services Contributing Writer. “IDS-Can You Afford Not To Have One?“. 02 Jun 2003. <http://www.networknewz.com/networknewz-10-20030602IDSCanyouaffordnottohaveone.html>

Deleted: → - → -

Deleted: - -

infrastructure and the 4250 is installed with the spanned infrastructure. The primary difference with these IDSs models is the MBS capability of processing network packets against signatures. Either model can be selected based on the amount of current and future calculated network throughput of an existing network infrastructure. Cisco performance numbers are based off specific benchmarking tests also noted in reference 1 below.

> The Cisco IDS 4250 supports superior performance at 500 Mbps and can be used to protect gigabit subnets and traffic traversing switches that are being used to aggregate traffic from numerous subnets. At 250 Mbps, the Cisco IDS 4235 can be deployed to provide protection in switched environments, on multiple T3 subnets, and with the support of 10/100/1000 interfaces, it can also be deployed on partially utilized gigabit links.[2]

To round out the high end 4200 series Cisco IDS, there is also an 4250-XL model which is capable of processing up to 1000mbs of data. This model will not be used in the below network environments. The minimum memory requirement for both 4235 and 4250 is 512mb. The hard-disk size is negligible because all alert collection will be sent to a central repository backed up on a SAN. Both models can be updated periodically with signature updates and tuning using Cisco Works.[3]

Special features of the Cisco IDS models include TCP connection resets, protection against IDS evasion techniques such as TCP and fragmentation reassembly, all of which are typical in most IDS systems. Additionally, both appliance based hardware devices are able to create extended ACLs to push to other Cisco devices. Cisco uses their IDS Active Response System to automatically block intrusions on other network devices such as routers, switches and firewalls.[4]

Network Architecture Diagram(s)

The below network diagrams will describe how network traffic is collected and sent to the IDS. For an initial implementation with cost restraints, the goal is to insert the IDS in a location where the largest amount of network traffic can be collected without upgrading existing network hardware.

Tier I site contains centrally located business infrastructure such as a data center. Remember, this is only an initial NIDS implementation. There are some

[2] Cisco Systems. "Cisco Intrusion Detection System Appliance and Module Installation and Configuration Guide Version 4". 2003. <http://www.cisco.com/application/pdf/en/us/guest/products/ps5398/c1676/ccmigration_09186a00801a24ce.pdf>

[3] Cisco Systems. "Release Notes for Cisco Secure Policy Manager Version 2.3.3i". Jan 2002. <http://www.cisco.com/en/US/products/sw/secursw/ps2133/prod_release_note09186a00800d9cc2.html>

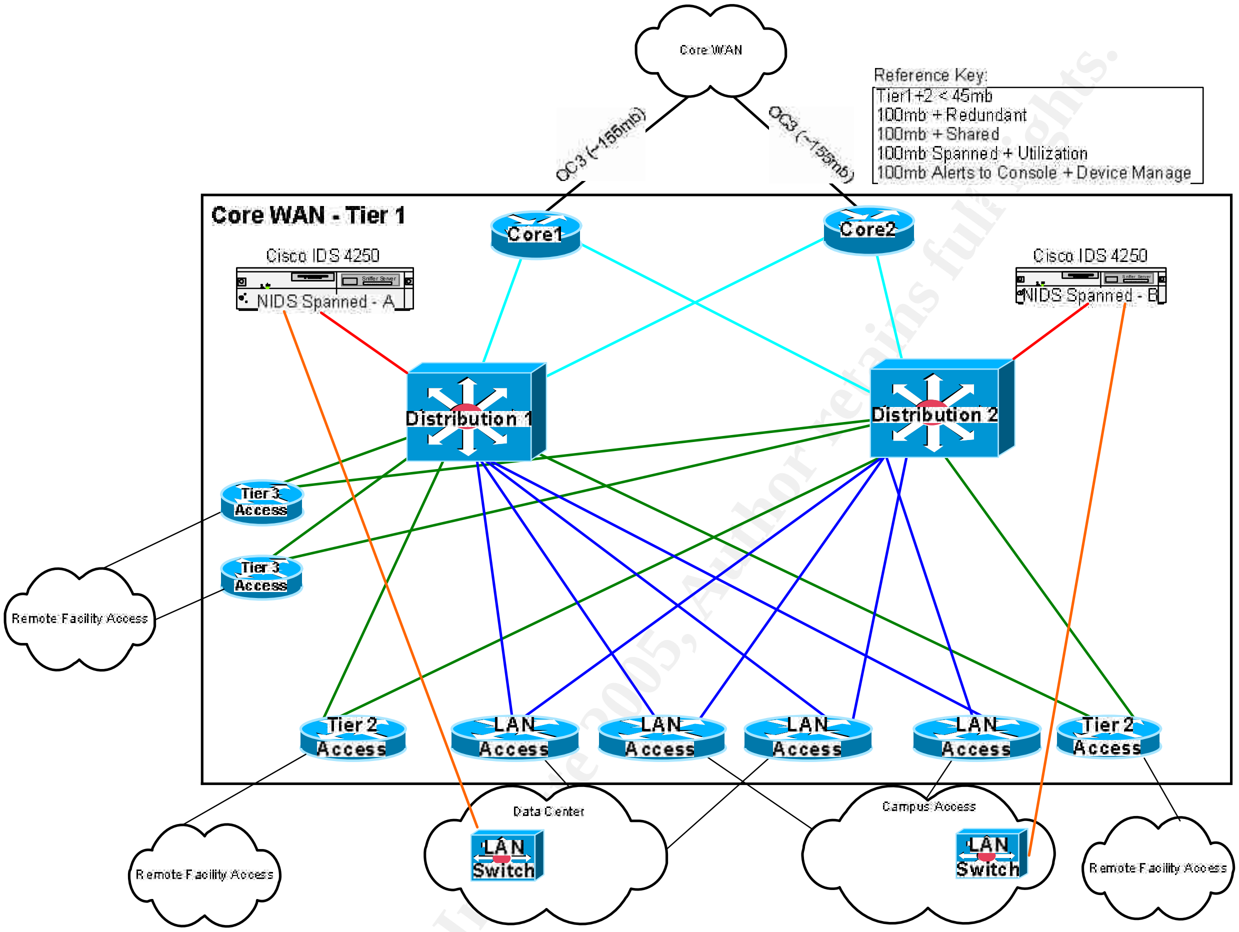
[4] Cisco Systems. "CISCO IDS 4200 SERIES SENSORS". 2004. <http://www.cisco.com/en/US/products/hw/vpndevc/ps4077/index.html>

Deleted: → - → -

Deleted: - -

inevitable weaknesses of the below NIDS infrastructure, which will be addressed with options for improvement below. There are also network architecture improvements which are outside the scope of this paper.
Tier I includes:

- No passive taps connected
- 2 Cisco 4250 NIDS spanned
- 1 addressable interface on each NIDS passes alerts back to the alert console



Data Flow:

In the diagram above connections feed into the multi-layer route\switches (a switch with a MSFC card) from the campus LAN, Data Center, Tier II\3, and other Tier I WAN facilities. This architecture is a result of network segmentation efforts following multiple virus outbreaks and can be considered one of many layer 3

Deleted: → - → -

Deleted: - -

(network layer) defense-in-depth strategies. Users connected to the campus network wanting to communicate with the data center must traverse thru the distribution switch; they cannot route directly. Same routing scenario applies to all network clouds outside of the perimeter of diagram 1. Each LAN router and Remote Facility can be considered isolated subnets.

From a routing perspective this provides an opportunity for network isolation. For example, if there is a virus outbreak within the campus access portion of the above network, it can be isolated from the rest of the corporate network very quickly and prevent further infection. Thus, only impacting the campus network and allowing the rest of the corporation can function normally .

From a security perspective these distribution switches provide an excellent ROI to install a Network Intrusion Detection System. All network traffic inbound or outbound to a Tier I facility and inbound or outbound between isolated subnets can be captured by an IDS. Since a majority of the corporate network traffic traverses these two distribution switches, all traffic can be sent to a spanned port with a NIDS device connected , as shown in the diagram above .

```
Config:
Distribution#1 sh run | inc monitor

Monitor session 1 source vlan 105 rx
Monitor session 1 destination interface Fa5/48

Interface status:
Distribution#1 sh int Fa5/48

(excerpt)
FastEthernet5/48 is up, line protocol is down (monitoring)
   5 minute input rate 0 bits/sec, 0 packets/sec
   5 minute output rate 96079000 bits/sec, 24636 packets/sec
```

Note in the below command, the interface is not assigned an IP address ("unassigned"). This output will be referenced in Detect One of Part II.

```
Distribution#1 sh ip int brief
excerpt) FastEthernet5 /48  unassigned  YES NVRAM  up   down
```

The indented quote limitations are eliminated with the above configuration. The config is to monitor a vlan and not specified ports. Every port on the switch assigned to vlan 105 above will be forwarded out Fa5/48 by the switch.[5] This reduces the amount of overhead needed when adding new connections to the distribution switches. For network technicians this should be similar to plugging in a network sniffer and configuring the switch to capture traffic . Furthermore,

[5] Cisco Systems. <http://www.cisco.com/en/US/customer/products/hw/switches/ps708/products_configuration_guide_chapter09186a008007f4c4.html>

Deleted: → - → -

Deleted: - -

additional vlans can be added to the session 1 monitor if needed. Concerning zero input rate traffic displayed on the interface status above "The alarms are sent out a separate management interface so as not to impede continual packet capture by the monitoring interface." [6]

Tier 1 architecture short falls:

There is an inherent network traffic limitation problem with this architecture. In the above network diagram there are six 100MB connections combined with four potentially 45MB connections and feeding their network traffic into each distribution switch, but only one 100MB (Fa5/48 above) interface. If we do the math, this is 6x100mb + 4x45m b = 780mb of network traffic and spanning it to a single 100mb port. A good example metaphor for this situation can be described as taking an 8 lane freeway and try ing to make them all get off a one lane off - ramp at the same time. Below is a more d etailed look at this scenario and allows us to get a good idea of how over utilized the spanned port may be in our Tier I architecture.

Cisco IDS 4250 - A:

```
Distribution1 # clear counters fa5/48
Distribution1# sh int fa5/48

(excerpt)
FastEthernet5 /48 is up, lin e protocol is down (monitoring)

Last clearing of "show interface" counters 00:05:00
5 minute output rate 96471000 bits/sec, 22195 packets/sec
Total output drops: 2,336,723
6,677,904 packets output
```

If we use the logic:

Total output drops + packets output = total traffic (offered to that interface)
Total output drops \Total traffic = %loss network traffic feeding into NIDS

Packet loss = 25.921%

Cisco IDS 4250 - B:

```
Distribution2# clear counters fa5/48
Distribution2# sh int fa5/48

(excerpt)
FastEthernet5/4 8 is up, line protocol is down (monitoring)

Last clearing of "show interface" counters 00:05:00
```

[6] Cisco Systems.
<http://www.cisco.com/en/US/customer/products/hw/vpndevc/ps4077/products_qanda_item09186a008017f8e4.shtml>

Deleted: → - → -

Deleted: - -

```
   5 minute output rate 96794000 bits/sec, 20172 packets/sec
   Total output drops: 2,332,353
   6,234,174 packets output
```

If we use the logic:
Total output drops + packets output = total traffic (offered to that interface)
Total output drops \Total traffic = %loss network traffic feeding into NIDS

Packet loss = 27.226%

The NIDS sensor can handle a much larger maximum bps than what is being offered by the 100mb output interface on the Distribution switches and the 4250 is far from being the bottleneck in this implementation. This design is currently dropping sniffed network packets at the Distribution switches. A network intrusion detection analyst will have a very difficult time analyzing signature alerts from Cisco IDS sensors A and B because the sensors are not receiving 100% of the network traffic. The sensors will not fire signatures on packets they never receive. Also, IDS evasion techniques have a much higher chance of success. The more fragments a sensor must reassemble the more likely one of those packets will be lost and the full packet reassembly will never occur, thus the signature will not fire to the alert console. To correct the packet loss and receive 100% accurate alert logs, there is a need to span the traffic to a Gigabit interface on each distribution switch. Alternatively, "The Catalyst 6000 IDS Module was designed specifically to address switched environments by integrating the IDS functionality directly into the switch and taking traffic right off the switch backplane."[7] Purchasing this new piece of hardware claims to resolve the above problem.

There is a second flaw to this architecture. The IDS will not be able to detect network traffic transmitted internally within each of the isolated subnets. The traffic will not traverse either Distribution switch; therefore it will not pass through the NIDS.
All this being said and assuming similar hardware costs, why wasn't a WS-SVC-IDSM2 switching module considered in the above network architecture? The WS-SVC-IDSM2 IDS module inserts just like a port module on a switch and directly sniffs packets from the backbone. [8] There are several pros and cons to an internal IDS blade vs. a external IDS appliance which is outside of the scope of this paper. Most likely the reasoning for the external NIDS device in a large organization is related to separation of duties. In most large organizations, the

[7] Cisco Systems. "CISCO IDS 4200 SERIES SENSORS". 2004. <http://www.cisco.com/en/US/products/hw/vpndevc/ps4077/products_qanda_item09186a008017f8e4.shtml>
[8] Cisco Systems. "Cisco Intrusion Detection System Appliance and Module Installation and Configuration Guide Version 4". 2003. <http://www.cisco.com/en/US/products/sw/secursw/ps2113/products_installation_and_configuration_guide_chapter09186a008014a238.html>

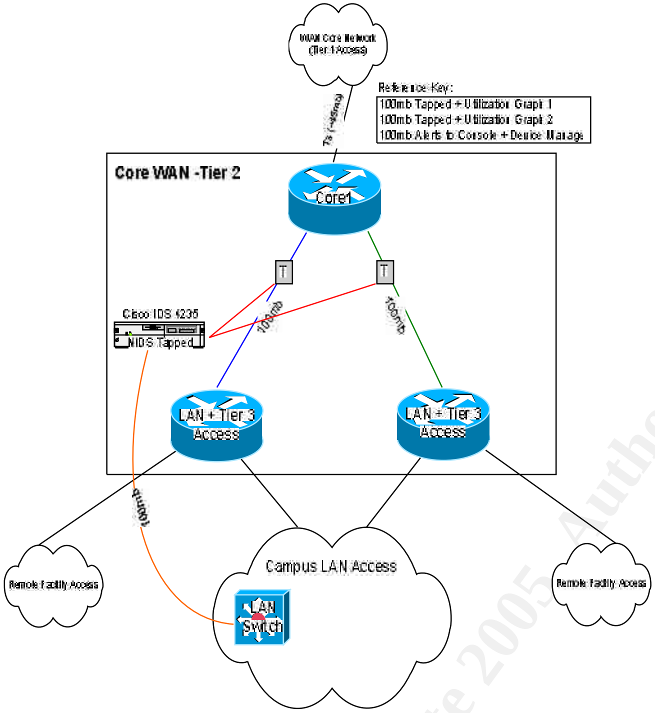
Deleted: → - → -
Deleted: - -

network engineer and the security engineer are not always the same person or organization.

Tier 2 – Remote large Sites
Tier II site contains critical distributed business infrastructure. The insertion of a NIDS device into this architecture uses passive taps.

Tier II includes:

- 2 passive taps
- 1 Cisco 4235 NIDS with 2 sniffing interfaces
- 1 addressable interface on each NIDS passes alerts back to the alert console



<u>Data Flow:</u>

Network traffic traversing inside and outside the remote facilities and campus LAN across the T3 circuit will be captured by the NID S devices in this architecture. N etwork traffic sourced or destined from the Campus LAN and Tier III facilities outside their isolated subnets will be collected by the NIDS, with the exception of Tier III to Tier II campus LAN access .

Deleted: → - → -

Deleted: - -

Two passive taps are inserted between the core router and the LAN routers. Additionally, two interfaces on the Cisco 4235 are configured to sniff network traffic. Taps "can send traffic data to the monitoring device by splitting or regenerating the network signal. Neither splitting nor regeneration introduce delay, or change the content of the information packets" [9] The GCIA material book 3.5\3.6 chapter 4 refers to information on passive taps can be found for Snort installations at www.snort.org. However, there is a significant difference in the way Cisco NIDS is tapped versus the snort diagram originated by Jeff Nathan at http://www.snort.org/docs/100Mb_tapping1.pdf.[10] Snort is primarily run on a server platform, while Cisco NIDS is considered a network appliance and more closely resembles a traffic analyzer (or piece of network equipment). In contrast to the above linked diagram, the Cisco NIDS runs a single port tap with a 100mb cable directly into the sniffing interface of the Cisco NIDS. The Cisco NIDS processes all packets against it signatures then sends alerts to the alerting console. Essentially, the biggest difference is the function of the 100mb Ethernet switch becomes integrated into the Cisco NIDS.

In the Tier II architecture there are two 100mb interfaces feeding the Cisco NIDS network traffic in comparison to our Tier I architecture where only one 100mb interface is transmitting network traffic. Because passive taps are used no configuration changes are required on the existing network equipment . Unlike our above spanned port, all interfaces on the Core and LAN routers have an assigned IP address. This means we can use a variety of known network monitoring tools to graphically represent network utilization traversing between the Core1 and Access routers. This traffic should be precisely the same traffic being sent to the Cisco NIDS. One such tool which can show the amount of data being sent to the Cisco 4235 NIDS is VitalSuite. This tool "…provides unsurpassed visibility into your infrastructure, letting you monitor, validate and enhance every aspect of your IT operations — helping you improve the quality of services you deliver to your users and capitalize on your resource investments." [11]

Green 100mb link

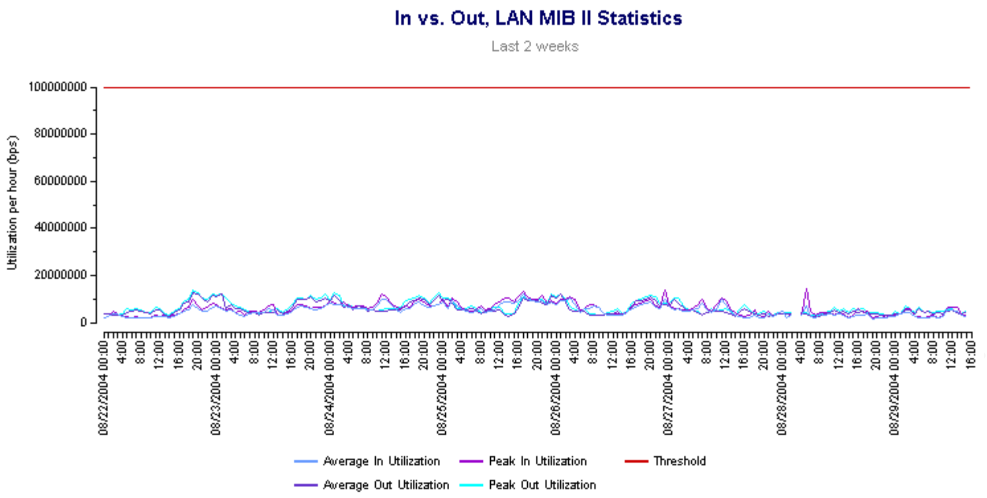
[9] Fischer, Amy. "Network taps enable passive monitoring". 28 Oct 2002. <http://www.nwfusion.com/news/tech/2002/1028techupdate.html>

[10] Nathan, Jeff and Caswell, Brian. 100Mb IDS Tapping Diagram (with only 100bt span port). <http://www.snort.org/docs/100Mb_tapping1.pdf>

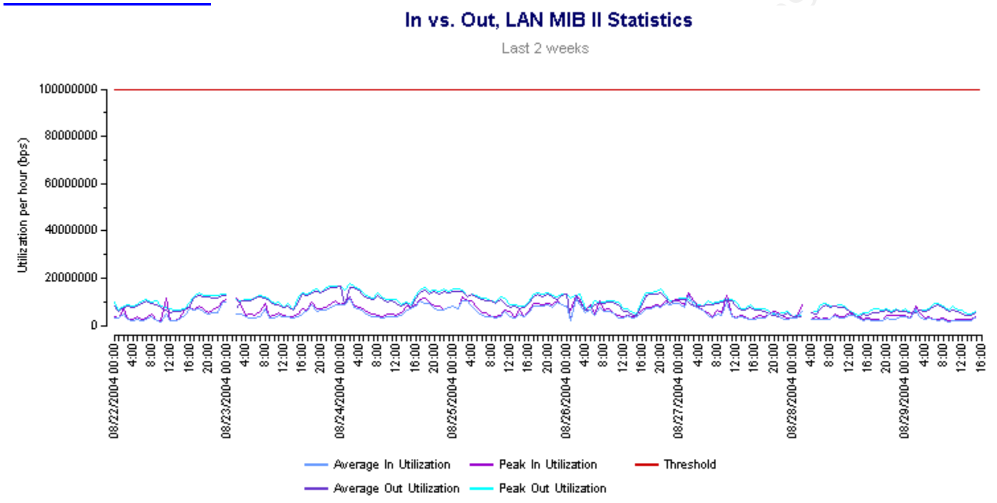
[11] Lucent Technologies. "Award-winning, performance-proven cost-saving software" <http://www.lucent.com/solutions/netops_enter.html>

Deleted: → - → -

Deleted: - -



Blue 100mb link



Unlike the traffic flow constrain ts detailed with Tier I, the Tier II architecture has room for growth. As shown above the 100mb taps are passing under 20mb of network traffic to the single Cisco 4235 NIDS.

There our two downsides to the way we have inserted the IDS into this architecture. First, n etwork traffic routed internally within the campus LAN, traffic routed internally within the Tier III facilities, and traffic routed between the Tier III facilities and Tier II campus LAN will not pass through the NIDS. This is missing more network traffic than in the Tier I design. The second downside does not deal with data collection, but rather the sustainabl e support of the infrastructure. Instead of routers and cables we now have a router, cable, tap, cable, router.

Deleted: → - → -

Deleted: - -

The tap, another piece of equipment, has been inserted into the direct path of network traffic. T his piece of equipment can fail, which will not only cause the NIDS to not be able to sniff network traffic, but will also cause a network service interruption.

Note:
The primary function of the non -sniffing IP addressable interface of the Cisco IDS is to transmit alerts to the analysis console. R emote management such as pushing new signatures and troubleshooting potential issues are other necessary functions of the addressable interface.

Sensor(s), Console(s) and Alert Collection

Keeping with the length requirements of the administrivia this pape r will not go into Cisco signatures or how they are updated . This is a gap and a list of Cisco signatures can be found here:
http://www.cisco.co m/en/US/products/sw/secursw/ps2113/products_data_sheet09186a008014c532.html

One method of updating Cisco signatures can be found here:
http://www.cisco.com/en/US/products/sw/cscowork/ps3991/products_user_guide_chapter09186a008018d96f.html

At this point, the network traffic has been collected, sent to the NIDS device , and processed against Cisco signatures. What will be done with the a lerts which match against the signatures? Assuming an enterprise corporation may have 5 Tier I sites and 20 Tier II sites, the WAN infrastructure would be composed of 30 NIDS. Remotely managing 30 separate NIDS systems individually would not be time or cost effective. A central logging mechanism is needed. Cisco provides a central logging mechanism which is a plug-in to Cisco works denoted as VMS . VMS provides data aggregation for a variety of Cisco networks devices, but “does not have any of the corre lation capabilities found in products from companies such as ArcSight, GuardedNet and netForensics.”[12] At this point, deviation from a stand alone Cisco IDS solution is essential to a large corporation. An advanced data correlation analysis tool, such as ArcSight, will improve efficiencies in finding real Events Of Interest.

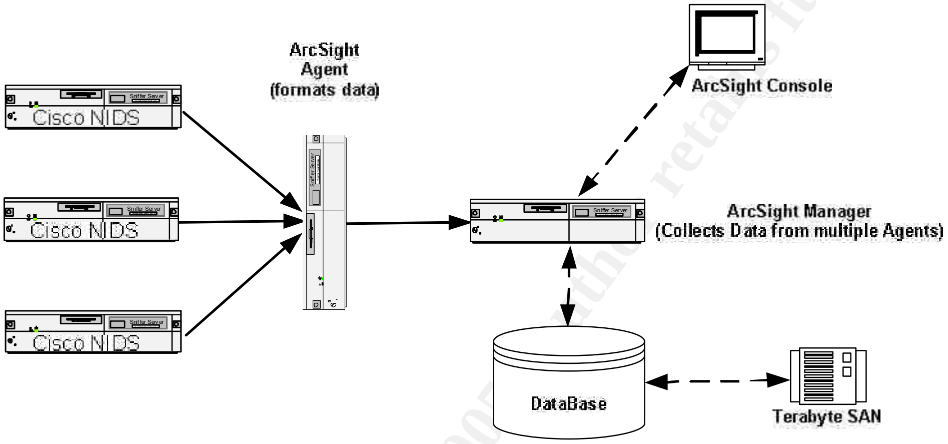
> ArcSight is a leading provider of enterprise software solutions that enable large organizations to better manage their security operations by integrating and optimizing the management of diverse security devices deployed across a network. By delivering complete aggregation, correlation, investigation, resolution and reporting…[13]

[12] Shipley, Greg and Miller, Patrick. “Cisco's NIDS Solution Grows Up”. 21 Oct 2002. <http://www.networkcomputing.com/1322/1322sp3.html>
[13] ArcSight. “Security Information Management Software for the Enterprise”. <http://www.arcsight.com>

Deleted: → - → -

Deleted: - -

ArcSight is built upon a multi-tier architecture involving SmartAgents, Managers, and Consoles.[14] In reference to the Tier I and II network infrastructures above, the Cisco NIDS will send alerts to a designated SmartAgent. The SmartAgent will then format the Cisco alert into a data format that will be understandable by the ArcSight manager. The ArcSight manager collects alerts sent from all SmartAgents and transmits the data to the user console in real time and a backend database for future analysis. The user console can be used to filter and sort alerts, create graphical displays such as charts, grids and reports, and query the database for past alerts. The database holds the alerts for 4 weeks on a terabyte storage area network (SAN). The alert traffic flow for the above architecture is graphically represented below. This is a self created graphical representation, however, similar flow diagrams from netForensics and GuardedNet (competitors of ArcSight) can be located at http://www.networkcomputing.com/1307/1307f22.html .


ArcSight Agent (formats data)
ArcSight Console
Cisco NIDS
Cisco NIDS
Cisco NIDS
ArcSight Manager (Collects Data from multiple Agents)
DataBase
Terabyte SAN

To provide the most efficient use and uninterrupted service, this entire alerting architecture should be replicated. For legal issues and concerns, the SAN should be capable of retaining and securely transferring unaltered alert information for a maximum of 30 days. Data passed between all above components is authenticated and encrypted thru the use of IPSEC software encryption using 3-DES.

For an initial installation, this IDS infrastructure was only setup to accommodate alerts from Cisco NIDS devices. For future expansion several recommendations can be made. To maximize ArcSight's data collection and correlation capabilities firewalls logs, router ACL logs, HIDS alerts, critical Server logs, ect. can be gathered thru additional SmartAgents. Once all core devices are correcting

[14] Christiansen, Chris and Kolodgy, Charles. "ArcSight Vendor Profile: Seeing Through the Clutter". Feb 2002. <http://www.arcsight.com/graphics/news/updated%20IDC%20report.pdf>

Deleted: → - → -

Deleted: - -

Field Code Changed

logging to ArcSight, the ArcSight application can be used to "ease the workload and increase the efficiency of overburdened security teams."[15]

Implementing this type of centralized alert collection architecture can lead to more efficiently finding Events of Interest in a large corporate environment. "We can't imagine running a modern-day SOC without the functionality they provide. In fact, we question the sanity of further IDS spending without correlation."[16]

Concept of Operations

With the above IDS model in place a centralized 24x7 Security Operations Team can provide the ability to monitor alerts across a global WAN environment, plus notify and contain events which may threaten the corporation. Given the above IDS infrastructure, with ArcSight integration, the SOC should be capable of protecting, detecting, and responding to incidents which may threaten or negatively impact the ability to do business within the corporate environment.

Monitoring Methodology and Procedures

Ideally, two security support specialist highly familiar with the corporate computing infrastructure should be attentive of the ArcSight Console at all times. Due to the sheer number of alerts being collected, a SOC analyst cannot base investigations solely on Red, Orange, Green alerts on the ArcSight console. An initial assessment will need to be prepared to determine the Event of Interest. After a preliminary assessment is complete, an analyst should be able to determine if the alert/s fall into one of the below 7 categories:

Malicious Hacker Activity
Malware Outbreak
New Malware Variant
Denial of Service Attacks
Unauthorized Scanning

Security Policy Violations
Mis-configured System
Alert Tuning Recommendation

Everything above the line will be escalated to a 2nd or 3rd level IT Security. Once escalated the preliminary investigation will be reviewed, revised and/or scrutinized immediately. The three items below the line are considered non -

[15] Janowski, Mike, Oele, Tom, and Shipley, Greg. "Too Much Information " 12 Sep 2003. <http://nwc.securitypipeline.com/showArticle.jhtml;jsessionid=K1PQCNLF2TWFCQSNDBGCKHQ?articleId=14700464&printableArticle=true>

[16] Janowski, Mike, Oele, Tom, and Shipley, Greg. "Too Much Information " 12 Sep 2003. <http://nwc.securitypipeline.com/showArticle.jhtml;jsessionid=K1PQCNLF2TWFCQSNDBGCKHQ?articleId=14700464&printableArticle=true>

Deleted: → - → -

Deleted: - -

urgent. A preliminary assessment will be constructed by the SOC analyst then sent for review by the 2 nd level personal.

Detect One [Truncated TCP OPTIONS]

2004-09-29 14:34:52 SID:1 CID:15557
(snort_decoder): Truncated Tcp Options
[TCP] 165.196.153.26:2161 -> 46.5.180.133:80
This Pure Secure output is only being used to quickly identify snort alerts and find an Event of Interest.

Source of Trace

Next is to determine which raw log file this detect came from using windump. Using the –X will show the HEX with windump ASCI translation output of the packet. Additionally, –vv option highlights some fields in the packet, which should make the analysis a bit (no pun intended) easier. The packet was found in raw log file 2002.6.11 using the below command. [17]

```
wd -nr c:\snort\bin\2002.6.11 -X -vv src host 165.196.153.26 and
dst host 46.5.180.133

20:13:00.824488 IP (tos 0x0, ttl 101, id 27440, len 317)
165.196.153.26.2161 > 4
6.5.180.133.80: S [bad TCP cksum e182 (->c441)!]
2543516742:2543517011(269) win
8338 <[bad opt]> (DF)bad cksum 8e27 (->8821)!

0x0000   4500 013d 6b30 4000 6506 8e27 a5c4 991a
E..=k0@.e..'....
0x0010   2e05 b485 0871 0050 979a fc46 0000 0000
.....q.P...F....
0x0020   7002 2092 e182 0000 4745 5420 2f69 6d61
p.......GET./ima
0x0030   6765 732f 736d 6275 6c6c 6574 2e6a 7067
ges/smbullet.jpg
0x0040   2048 5454 502f 312e 310d 0a41 6363 6570
.HTTP/1.1..Accep
0x0050   743a 202a 2f2a 0d0a 5265 6665 7265 723a
t:.*/*..Referer:
0x0060   2068 7474 703a 2f2f 7777 772e 5858 5858
.http://www.XXXX
0x0070   2e63 6f6d 2f6d 6169 6e2f 6361 7461 6c6f
.com/main/catalo
0x0080   672f 6c61 6e39 3163 3131 312e 6874 6d6c
g/lan91c111.html
0x0090   0d0a 4163 6365 7074 2d4c 616e 6775 6167        ..Accept-
Languag
0x00a0   653a 2065 6e2d 7573 0d0a 4163 6365 7074        e:.en-
us..Accept
0x00b0   2d45 6e63 6f64 696e 673a 2067 7a69 702c        -
Encoding:.gzip,
```

[17] <http://isc.sans.org/logs/raw/ file 2002.6.11>

Deleted: → - → -

Deleted: - -

```
0x00c0   2064 6566 6c61 7465 0d0a 5573 6572 2d41        .deflate..User-
A
0x00d0   6765 6e74 3a20 4d6f 7a69 6c6c 612f 342e
gent:.Mozilla/4.
0x00e0   3020 2863 6f6d 7061 7469 626c 653b 204d
0.(compatible;.M
0x00f0   5349 4520 362e 303b 2057 696e 646f 7773
SIE.6.0;.Windows
0x0100   2039 383b 2054 3331 3234 3631 290d 0a48
.98;.T312461)..H
0x0110   6f73 743a 2077 7777 2e58 5858 582e 636f
ost:.www.XXXX.co
0x0120   6d0d 0a43 6f6e 6e65 6374 696f 6e3a 204b
m..Connection:.K
0x0130   6565 702d 416c 6976 650d 0a0d 0a               eep-Alive....
```

Now that the source raw log file (2002.6.11) has been uncovered we can start to determine how the packet was captured by the Snort IDS. This portion of analysis follows some of the techniques used by Peter H. Storms GCIA Honors practical.[18] Using the following commands -n for no name resolution (speeds up processing time), -e for displaying the Ethernet frame headers, and –r to read the set designated file; we can determine the source and target hardware addresses of detect#1 packet.

```
wd -ner c:\snort\bin\2002.6.11 dst host 46.5.180.133
wd -ner c:\snort\bin\2002.6.11 src host 165.196.153.26
```

The commands result in the below MAC address designations. The IDS is between these 2 devices.

```
0:3:e3:d9:26:c0 0:0:c:4:b2:33
0:0:c is Cisco
0:3:e3 is Cisco[19]
```

This shows the packet was passed between 2 unique Cisco based network interface cards. External information about the 2 unique Ethernet addresses from www.cisco.com insinuates designations to the particular hardware devices. Searching for "00-00-0c-40" brings up documentation only related to catalyst switches. Whereas, MAC address "00-03-e3" appears to be a specifically related to a "CISCO UBR7200 SERIES UNIVERSAL BROADBAND ROUTERS" along with how to configure the router for a LAN Sniffing Device such as an IDS.[20]

To confirm we do not have another IDS system out there logging to the same 2002.6.11 file, we can confirm with the below command:

[18] Storm, Peter H. "GIAC Certified Intrusion Analyst (GCIA) Practical Assignment Version 3.3". 15 Nov 2003. <http://www.giac.org/practical/GCIA/Pete_Storm_GCIA.pdf>

[19] IEEE. 2004. <http://standards.ieee.org/regauth/oui/oui.txt>

[20] Cisco Systems. <http://www.cisco.com/en/US/customer/products/hw/cable/ps2217/products_feature_guide_chapter09186a008019b571.html>

Deleted: → - → -

Deleted: - -

```
wd -ner c:\snort\bin\2002.6.11 ether src not 0:0:c:4:b2:33 and
ether dst not 0:0:c:4:b2:33
```

No results are returned, so it safe to say that raw log file 2002.6.11 came from only one IDS sensor placed between a Cisco Router and Cisco Catalyst Switch. Again this only gives an idea of the infrastructure on which the above packet was captured.

Returning to windump and issuing ether dst and ether src commands will give a better understanding of the data transversal in this network.

```
wd -ner c:\snort\bin\2002.6.11 ether src 0:0:c:4:b2:33 >
c:\ryan\gcia\part2\IP1
```

Only 2 source IP addresses are coming from IP1 file output results:

```
46.5.180.250
46.5.180.133
```

```
wd -ner c:\snort\bin\2002.6.11 ether dst 0:3:e3:d9:26:c0 >
c:\ryan\gcia\part2\IP4
```

The 2 same source IP addresses are coming from IP4 file output results:

```
46.5.180.133
46.5.180.250
```

Both IP addresses seem to be very active in the above windump filters. Additionally, the 46.5.180.133 address is the same as the target address in Detect#1. Initially, I thought these 2 IP addresses might be assigned to each of the below Ethernet addresses. However, after running the next windump filter with the below output this cannot be determined. The target host above does reply to port 80 requests from other source hosts . A router or switch typically does not respond to port 80 requests, they only forward or route packets on there way. Without having access to the network equipment, it is inconclusive to say the 46.5.180.133 or 46.5.180.250 IP addresses are assigned to the Hardware addresses shown above.

```
wd -nr c:\snort\bin\2002.6.11 src host 46.5.180.133 >
c:\Ryan\GCIA\Part2\Source\Does_target_reply2

21:53:56.074488 IP 46.5.180.133.80 > 147.91.1.45.35343: P
3574766657:3574767204(547) ack 3241705904 win 32120 (DF)
21:53:57.154488 IP 46.5.180.133.80 > 147.91.1.41.4984: P
3571297796:3571298343(547) ack 3506514742 win 31856 <nop,nop,timestamp
283935 1822338> (DF)
21:53:57.724488 IP 46.5.180.133.80 > 147.91.1.41.4991: P
3573807056:3573807603(547) ack 3508250192 win 31856 <nop,nop,timestamp
283993 1822446> (DF)
```

Deleted: → - → -

Deleted: - -

```
23:29:22.744488 IP 46.5.180.133.80 > 195.29.208.9.47537: P
1038045389:1038045931(542) ack 968323269 win 31856 <nop,nop,timestamp
856420 231250163> (DF)
23:29:25.844488 IP 46.5.180.133.80 > 195.29.208.9.47614: P
1045041376:1045041918(542) ack 971804608 win 31856 <nop,nop,timestamp
856729 231250708> (DF)
09:34:52.924488 IP 46.5.180.133.80 > 213.191.135.229.1381: P
748644380:748644916(536) ack 21048548 win 32696 (DF)
```

The above output and below output excerpt uncovers network traffic flows. From this information an internal network address space can be defined.

```
(output excerpt)
IP 46.5.180.250.61982 > 64.154.80.50.80
IP 46.5.180.133.80 > 213.191.135.229.1381
```

Before a network diagram is constructed displaying the source of Detect#1 , the below windump filter can show if there are any additional networks. The filter results also show that the external internet traffic is inbound to the 0:0:c:4:b2:33 hardware device and to 46.5.80 subnet space.

```
wd -ner c:\snort\bin\2002.6.11 ether dst 0:0:c:4:b2:33 >
c:\ryan\gcia\part2\IP2
```

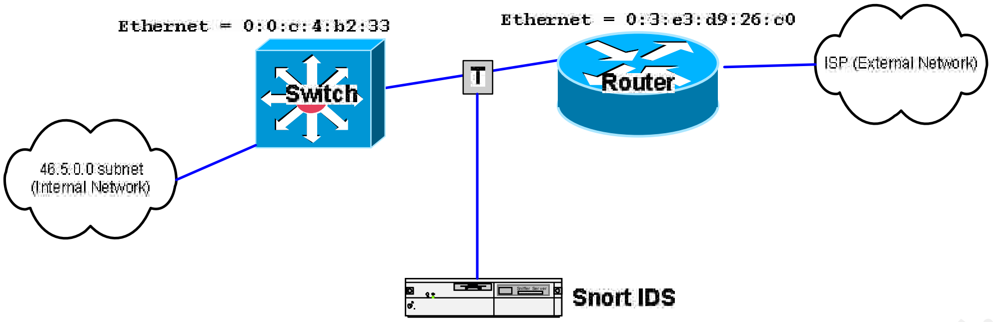
```
(output excerpt)
IP 66.125.147.222 > 46.5.80.149
IP 12.99.244.2.80 > 46.5.80.149
IP 64.3.83.34.80 > 46.5.80.149
IP 65.113.31.2.80 > 46.5.80.149
IP 206.111.234.194.80 > 46.5.80.149
```

The output from the IP2 file is not very large (~89K). If the file contained thousands of records then an alternative method such as using awk or sort and uniq –d unix based commands may have been employed. However, the exact number of source hosts is not needed. A quick scan of the IP2 output file should suffice for discovering any additional networks. The IP2 file contains 130+ hosts in the 46.5 address space, which are targets in the 2002.6.11 log file. This internet entity has been allocated a 46.5.0.0/16 subnet. No other networks are used in raw log file 2002.6.11.

Based on this analysis the network should look similar to this assuming a tapped infrastructure:

Deleted: → - → -

Deleted: - -



Detect Generation Method

Downloaded every file from http://isc.sans.org//logs/raw/ on or before September 21, 2004, place them in a batch file, ran them through snort 1.9, disabled the preprocessors, output them to a mysql database, and displayed the alert output through PureSecure 1.6 front -end. Essentially, the standard installation of PureSecure version 1.6, which installs snort with fnord, arpspoof and arpspoof_detect_host preprocessors d isabled within the snort.conf.

"(snort_decoder): Truncated TCP Options" alert was found using the below snort command. The 2002.6.11 file was read in using the –r option and was processed against the snort1.conf f ile installed with PureSecure1.

```
c:\PureSecure \bin\snort -r C:\Snort\bin\2002.6.11 -c
conf\snort1.conf
```

The only reference to the generation of this alert in the snort.conf file states:

```
    # Snort's decoder will alert on lots of things such as header
    # truncation or options of unusual length or infrequently used TCP
    options
```

The alert can be disabled by removing the pound sign from the "# config disable_TCPopt_alerts"[21]

Since Detect#1 did not have a typical specified snort alert found in other detects[22], the below analy sis was performed to find how Detect#1 was generated. To determine what generated D etect#1, I issued the –vv for very verbose and –x for display of Hex output. The output is modified to show only the Hex code from the IP and TCP Headers.

```
wd -nvvxr c:\snort\bin\2002.6.11 src host 165.196
```

[21] procana@insight.rr.com. Neohapsis: Re: [Snort-users] (snort_decoder): Truncated Tcp Options. 27 Apr 2003. <http://archives.neohapsis.com/archives/snort/2003-04/1176.html>
[22] Snort Signature Database. "Building Networks on the Fly". 2004. <http://www.snort.org/snort-db/sid.html?sid=116-55>

Deleted: → - → -

Deleted: - -

```
20:13:00.824488 IP (tos 0x0, ttl 101, id 27440, len 317)
165.196.153.26.2161 > 46.5.180.133.80: S [bad TCP cksum e182 (->c441)!]
2543516742:2543517011(269) win 8338 <[bad opt]> (DF)bad cksum 8e27
(>8821)!

4500 013d 6b30 4000 6506 8e27 a5c4 991a
2e05 b485 0871 0050 979a fc46 0000 0000
7002 2092 e182 0000 4745 5420 2f69 6d61
```

Inspecting the output from the above command , it appears there are two checksum errors. The IP header bad checksum value appears to be larger. The detect generation method for this alert is shown with the output "bad cksum 8e27 (>8821)!" If the bad IP Header checksum is caused by the source and destination IP addresses being altered or corrupted, this will also generate a bad TCP header checksum pe r the pseudo -header protection TCP checksum.[23] These checksum errors could potentially generate Detect#1. By further decoding the packet, binary discrepancies should uncover the reason Detect#1 was generated.

IP Header				
4-bit version **Value=4**	4-bit header length **Value=5**	8-bit TOS **Value =00**	16-bit total length (in bytes) **Value=013d**	
16-bit IP identification number **Value=6b30**			R, DF, and MF 3-bits **Value=4**	13-bit fragment offset **Value=000**
8-bit Time to Live **Value=65**		8-bit protocol **Value=06**	16-bit header checksum **Value=8e27**	
32-bit source IP address **Value=a5c4:991a**				
32-bit destination IP address **Value=2e05:b485**				

Version - is normal with IPver4

Header length - is standard 5 Hex value. Telling us there is no IP options set in this packet or additional data set in this packet. Multiplying Header Length (5) by the Version (4) gives us a total IP header length of 20bytes.

TOS - is normal with No options set. There is no priority set on this packet.

IP identification field - uniquely identifies each datagram sent by the source host.

[23] Roesch, Martin and Poor, Mike. Track 3: Intrusion Detection In-Depth, Network Traffic Analysis using tcpdump. Parts 1 and 2 (slide 3-34). 2004.

Deleted: → - → -

Deleted: - -

Fragment Offset - is set to 0x40 which sets the DF bit or Don't Fragment Bit. Following the logic 0x40= 16^1*4 + 16^0 = 64. 64 sets the 6th bit in Binary (read right-to-left). Following the 128, 64, 32, 16, 8, 4, 2, 1 bit masking convention, the sixth bit is 64.

TTL - 0x65 = 16^1 *6 + 1 6^0 *5 = 101 TTL. Nothing unusual here. Several windows operating systems start their TTL values at 128. [24] However, assuming the source is windows the TTL has been decremented 128 -101= 27 times before reaching the source of 46.5.180.133. This is high, b ut possible. There is no evidence to suggest the TTL value has generated the alert

Protocol field - 0x06 signifying the packet contains data in a TCP format following the IP header. Due to the title of the detect " **Truncated Tcp Options"** , the TCP header, specifically the TCP options field is suspected to have generated the snort_decoder alert.

IP Header Checksum - this is larger than normal, but would not have indirectly triggered the snort_decoder [Truncated TCP Options] and fired the alert per the fields computed by the TCP checksum.[25] The IP and TCP checksums are not completely independent of one another. Essentially, if the IP checksum is incorrect, the TCP checksum computes some of the same IP header field values , and may or may not be inco rrect. If the IP checksum failed because the source or destination IP addresses are corrupted or sanitized before being posted to the internet, then the IP fields correspond to the same 2 fields the TCP checksum computes. Consequently, both checksums would fail.[26] Even though the IP header checksum failure should not trigger the snort_decoder[Truncated TCP Options] alert, below will show how the IP checksum failed.

Pulling up the raw log 2002.6.11 in ethereal states Header Checksum: 0x8e27 (incorrect, should be 0 x8821), same as the windump output above. Essentially, the IP Header checksum is showing more bits turned on. Taking the correct value 0x8821 and calculating into Decimal\Binary we see 8*16^1 + 8*16^0 = 136(Binary=10001000) and 2*16^1 + 1*16^0 = 33(Binar y=00100001). With the invalid IP Header Checksum of 0x8e27 where Hex e= 14 we see 8*16^1 + 14*16^0 = 142(Binary=10001110) and 2*16^1 + 7*16^0 = 39(Binary=00100111).

In binary comparison this is how the two compare:
0x8821 = 10001000:00100001
0x8e27 = 10001110:00100111

[24] The Swiss Education & Research Network. "Default TTL Values in TCP/IP". 2004. <http://secfr.nerim.net/docs/fingerprint/en/ttl_default.html>

[25] Kozierok, Charles M. 'TCP Checksum Calculation and the TCP "Pseudo Header"' Version 2.0. 7 Jun 2004. <http://www.tcpipguide.com/free/t_TCPChecksumCalculationandtheTCPPseudoHeader-2.htm>

[26] Kozierok, Charles M. 'TCP Checksum Calculation and the TCP "Pseudo Header"' Version 2.0. 7 Jun 2004. <http://www.tcpipguide.com/free/t_IPDatagramGeneralFormat.htm>

Deleted: → - → -

Deleted: - -

There are 2 additional bits flipped in each of the 2 bytes. This could be additional data inserted into the IP packet before the IP[10] byte offset (IP Header checksum) or a the IP packet could have been corrupted during transmission. However, before making these assumptions similar packets in the 2002.6.11 log file should be checked.

```
wd -nr c:\snort\bin\2002.6.11 -vv dst port 80 >
c:\ryan\gcia\part2\DSTPort80

(All records omitted - excerpt only)
19:29:15.334488 IP (tos 0x0, ttl 238, id 2723, len 264)
192.77.15.39.53447 > 46.5.180.133.80: P [bad TCP cksum d04e (->ea7e)!]
1985084966:1985085190(224) ack 3013725665 win 8760 (DF)bad cksum d553
(->cf4d)!
19:34:14.884488 IP (tos 0x0, ttl 44, id 55568, len 40) 202.96.52.99.80
> 46.5.206.112.80: . [bad TCP cksum b28e (->ad86)!] 253:253(0) ack 0
win 1400bad cksum bf8e (->ba86)!
19:34:19.984488 IP (tos 0x0, ttl 44, id 56082, len 40) 202.96.52.99.80
> 46.5.206.112.80: . [bad TCP cksum b22d (->ad25)!] 97:97(0) ack 1 win
1400bad cksum bd8c (->b884)!
19:34:24.914488 IP (tos 0x0, ttl 43, id 56622, len 40) 218.96.62.2.80 >
46.5.206.112.80: . [bad TCP cksum 9828 (->9320)!] 452:452(0) ack 0 win
1400bad cksum a2d1 (->9dc9)!
19:34:30.034488 IP (tos 0x0, ttl 43, id 57164, len 40) 218.96.62.2.80 >
46.5.206.112.80: . [bad TCP cksum 97c4 (->92bc)!] 100:100(0) ack 1 win
1400bad cksum a0b3 (->9bab)!
19:38:30.194488 IP (tos 0x10, ttl 240, id 0, len 1424)
62.22.119.112.2374 > 46.5.180.133.80: P [bad TCP cksum 0 (->a55e)!]
3599425360:3599426744(1384) ack 249591508 win 31740bad cksum 0 (-
>2d47)!
19:42:09.444488 IP (tos 0x0, ttl 46, id 42297, len 40) 163.23.190.2.80
> 46.5.112.165.80: . [bad TCP cksum af9a (->a795)!] 611:611(0) ack 0
win 1400bad cksum efd7 (->e7d2)!
19:53:51.044488 IP (tos 0x10, ttl 240, id 0, len 1504)
209.92.27.16.3469 > 46.5.180.133.80: P 262122120:262123584(1464) ack
683495803 win 32120bad cksum 0 (->f610)!
20:12:40.634488 IP (tos 0x0, ttl 124, id 10447, len 141)
46.5.180.250.61982 > 64.154.80.50.80: P [bad TCP cksum 49f9 (->43f3)!]
274655959:274656060(101) ack 3266065286 win 8760 (DF)bad cksum 67d6 (-
>61d0)!
20:12:40.644488 IP (tos 0x10, ttl 240, id 0, len 1500)
46.5.180.250.61982 > 64.154.80.50.80: P [bad TCP cksum 0 (->5aaf)!]
2991409327:2991410787(1460) ack 1303557970 win 33580bad cksum 0 (-
>5140)!
```

After running the above bpf filter with windump, the output shows all packets to the DSTPort80 file have bad cksum's. There was not a snort alert generated by every packet queried by this bpf filter. Based on these results, it is safe to say, a bad checksum (IP or TCP) is not what has generated the Detect#1 alert. The cause of these incorrect header checksums in the 2002.6.11 log file most likely occurred because true IP addresses have been obfuscated before being posted to http://isc.sans.org/logs/raw/. This logical analysis is further confirmed with the windump ASCI "http://www.XXXX" in the payload of Detect#1. The XXXX

Deleted: → - → -

Deleted: - -

appears to be substituted for the original IP address. Based on this analysis, both bad TCP and IP checksum annotations in the log file must be disregarded as anomalies.

The generation of the D etect lies exactly where the Detect title insinuates; within the TCP Options.

TCP Header								
16-bit source port number **Value=0871**		16-bit destination port number **Value=0050**						
32-bit sequence number **Value=979a:fc46** 32-bit acknowledgement number **value=0000:0000**								
4-bit header length **Value=70**	6-bits reserved	URG **0**	ACK **0**	PSH **0**	RST **0**	SYN **1**	FIN **0**	16-bit window size **Value=2092**
16-bit checksum **Value=e182**		16-bit urgent pointer						**Value=0000**
32-bit options **Value=4745:5420:2f69:6d61**								

Source Port – 2161. Nothing abnormal. No related Trojans, backdoors, ect. [27]

Destination Port – 80 (HTTP).

Sequence Number - seems normal.

Acknowledgement N umber - Acknowledgement number is 0. This seems odd until you look at the 13byte off set, where the flags are set. The SYN in th e only flag set. This packet appears to be an initial SYN packet.

Header Length - is 28 bytes long 7*16^1 = 112(Binary=01110000). A standard TCP header is only 20 bytes long. Options must be set. This warrants so me further analysis. Using more advanced windump bpf filters can show why this packet is unique. First, how many packets have options set?

```
wd -nr c:\snort\bin\2002.6.11 -vv "TCP[12] & 0xf0 > 0x50"
```

30+ packets with TCP options set. Having a TCPheader offset greater than 0x50 does not seem to be the cause of Detect#1. Next, how many additional packets have a 0x7 in the high order nibble of the 12th byte offset?

```
wd -nr c:\snort\bin\2002.6.11 -vv "TCP[12] & 0xf0 = 0x70"

20:13:00.824488 IP (tos 0x0, ttl 101, id 27440, len 317)
165.196.153.26.2161 > 46.5.180.133.80: S [bad TCP cksum e182 (->c441)!]
```

[27] DShield. Port Report: 80. 27 Oct 2004. <http://www.dshield.org/port_report.php>

Deleted: → - → -

Deleted: - -

```
2543516742:2543517011(269) win 8338 <[bad opt]> (DF)bad cksum 8e27
(>8821)!
```

This seems strange, there should be more packets with 0x70 set in the 12 byte offset of the TCP header. Maybe the 2002.6.11 log file does not have any TCP[12] matching packets. Before a determination is made, I decided to run all the downloaded files through the above TCPdump bpf filter. Several have matching TCP header lengths of 0x07 in the 12th byte offset, including file 2002.6.10 with 3 of the shown packets below:

```
wd -nr c:\snort\bin\2002.6.10 -vvx "TCP[12] & 0xf0 = 0x70"

02:39:52.784488 IP (tos 0x0, ttl 107, id 3035, len 48)
61.222.198.26.62402 > 46.5.180.251.8080: S [bad TCP cksum 5bac (-
>55a6)!] 4186962332:4186962332(0) win 16384 <mss 1460,nop,nop,sackOK>
(DF)bad cksum 22fa (->1cf4)!
                         4500 0030 0bdb 4000 6b06 22fa 3dde c61a
                         2e05 b4fb f3c2 1f90 f98f f99c 0000 0000
                         7002 4000 5bac 0000 0204 05b4 0101 0402
```

I checked PureSecure front end for any other triggered snort alerts on the above source address of 61.222.198.26. No alerts were found. This means the TCP header length of 0x70 did not trigger Detect #1 or any other snort alerts. However, there is something significantly different about the above packet verses the Detect#1 packet. Referring back to the neohapsis archives, Mike states a normal window scale looks like "... 02 04 05 b4 ... " in the TCP[20], [21], [22] and [23] byte offset.[28] If we run the below command matching the above Hex there is 3 matching entries, same as the previous TCPdump bpf filter.

```
wd -nr c:\snort\bin\2002.6.10 -vv "TCP[12] & 0xf0 = 0x70 and
TCP[20] = 0x02 and TCP[21] = 0x04 and TCP[22] = 0x05 and TCP[23]
= 0xb4"

02:39:43.804488 IP (tos 0x0, ttl 107, id 3004, len 48)
61.222.198.26.62402 > 46.5.180.251.8080: S [bad TCP cksum 5bac (-
>55a6)!] 4186962332:4186962332(0) win 16384 <mss 1460,nop,nop,sackOK>
(DF)bad cksum 2319 (->1d13)!
02:39:46.774488 IP (tos 0x0, ttl 107, id 3014, len 48)
61.222.198.26.62402 > 46.5.180.251.8080: S [bad TCP cksum 5bac (-
>55a6)!] 4186962332:4186962332(0) win 16384 <mss 1460,nop,nop,sackOK>
(DF)bad cksum 230f (->1d09)!
02:39:52.784488 IP (tos 0x0, ttl 107, id 3035, len 48)
61.222.198.26.62402 > 46.5.180.251.8080: S [bad TCP cksum 5bac (-
>55a6)!] 4186962332:4186962332(0) win 16384 <mss 1460,nop,nop,sackOK>
(DF)bad cksum 22fa (->1cf4)!
```

Last confirmation of Detect#1's generation method. I checked all the files where the 12 byte offset of the TCP header = 0x70, but the 20, 21, 22, and 23rd byte offsets do not equal 0x0204 05b4.

[28] procana@insight.rr.com. Neohapsis: Re: [Snort-users] (snort_decoder): Truncated Tcp Options. 27 Apr 2003. <http://archives.neohapsis.com/archives/snort/2003-04/1176.html>

Deleted: → - → -

Deleted: - -

Field Code Changed

```
wd -nr c:\snort\bin\2002.6.10 -vv "TCP[12] & 0xf0 = 0x70 and TCP[20] !=
0x02 and TCP[21] != 0x04 and TCP[22] != 0x05 and TCP[23] != 0xb4" >
c:\ryan\gcia\part2\OptionAnomallyCheck_2002.6.10
```

No output generated from any of the log files.

Based on the above analysis, Detect#1 has been generated because the TCP options field value equals 0x47455420 2f696d61 with a TCP header length of 28bytes (0x70). Whereas a normal TCP option value (with a the TCP header length of 28bytes) set equals 0x020405b4. Detect#1 has been generated because more than 8 bytes of data have been read into the options field.

Is this OS fingerprinting scanning activ ity? What does 0x47455420 mean? A gentle search in www.google.com for "0x47455420" results in the translation of 0x47455420 to equal "GET" [29]

The formula is a s follows:
TCP[((TCP[12] & 0xf0) / 4):4] = 0xFFFFFFFF

The "TCP[((TCP[12] & 0xf0) / 4)" says take the higher order nibble of the TCP[12] byte offset and divide it by 4. The ":4" states to read the next 4 bytes. "F" equals the specified Hex value. Any 8 byte value can replace the above F's to search for specific payload content, regardless of the TCP options set. For a further example 0x51554954 translates in ASCI to "QUIT". [30]

Lets verify the above is correct within the http://isc.sans.org/logs/raw/ 2002.6.11 file where Detect#1 was found.

```
wd -nr c:\snort\bin\2002.6.11 -vvX "TCP[((TCP[12] & 0xf0) / 4):4]
= 0x47455420" > c: \ryan\gcia\part2\Found_Cause
```

Found_Cause file results in 304KB of data. Looking at Found_Cause file verifies every packet contains a "GET" data in the payload. As a redundancy check I performed the below TCPdump bpf filter on all log files checking for anything with the 8 bytes of options set (0x70) and "GET" in the payload.

```
wd -nr c:\snort\bin\2002.6.11 -vvX "TCP[((TCP[12] & 0xf0) / 4):4]
= 0x47455420" and " TCP[12] & 0xf0 = 0x70"
```

No results from any log file, as expected. Based on this correlation, the first part of the payload resides in the 8 byte field reserved for the TCP options in Detect#1. Instead of having truncated options, there are actually NO options.

[29] Lindsey, Mark R. University of North Carolina, Department of Computer Science: Work Log: Wed Jan 29 10:57:08 EST 2003. 29 Jan 2003. <http://www.cs.unc.edu/~lindsey/7ds/log/>
[30] Bakos, George. TCPDUMP Public Repository: Re: [tcpdump-workers] understanding filtering. 17 Dec 2002. <http://www.tcpdump.org/lists/workers/2002/12/msg00088.html>

Deleted: → - → -

Deleted: - -

How did this happen? There are two possibilities on how this may have occurred. One, looking at the binary of the TCP header field for 2 packets, 1 with and 1 without options set shows:

20 byte packet length = 0x50 = 5*16^1 + 0*16 ^0 = 80
28 byte packet length = 0x70 = 7*16^1 + 0*16^0 = 112

128 64 32 16 8 4 2 1

01010000 = 80
01110000 = 112

The 3rd bit could have been corrupted causing the TCP header length to enable 8 bits of options and create a 28 byte TCP header length. Since no real options were specified in this packet, the data portion of the payload could have been read into the TCP header.

Second, the TCP header length could have been a legitimate value of 0x70. However, somewhere during data transmission of this packet , the 8 bytes of TCP options may have been dropped.

Address Spoofing Probability

Since we only have one packet which is not a broadcast address and the payload has been read into the options field, as noted above, the probability the source address is spo ofed is very low. Detect#1 above is a single syn packet possibly looking for a response or No Response. When TCP options are changed this is a common symptom of OS detection fingerprinting scanners. [31] Someone running nmap with the –O option on the source host could have generated the **(snort_decoder): Truncated TCP Options** alert.[32] If the source machine had the intention of doing OS fingerprinting, they would likely need a valid source ip address to interpret a response.

Furthermore, the TTL value is 101. Above I noted the initial TTL most likely is 128 and originated from a windows operating system (reference 14). 27 hops is a high hop count, but is possible. When tracing to the source address from England, the hop count is 19 hops before enter the Los Rios Community College District subnet address range in Sacramento, Ca. [33] All it would take is eight more routers inside the source or destination's LAN to bump the hop count up to 27.

```
Tracing the route to 165.196.153.26
```

[31] fyodor@insecure.org. "Remote OS detection via TCP/IP Stack FingerPrinting". 11 Jun 2002. <http://www.insecure.org/nmap/nmap-fingerprinting-article.html>

[32] Bauer, Mick. Linux Journal: Issue 85: Paranoid Penguin: Checking Your Work with Scanners, Part I (of II): nmap. 1 May 2001. <http://www.linuxjournal.com/article.php?sid=4561>

[33] American Registry for Internet Numbers. <http://ws.arin.net/cgi-bin/whois.pl>

Deleted: → - → -

Deleted: - -

```
  1 158.43.56.229 [AS 702] 0 msec 0 msec 4 msec
  2 158.43.145.33 [AS 702] 4 msec 0 msec 4 msec
  3 158.43.254.182 [AS 702] 12 msec 8 msec 12 msec
  4 158.43.254.149 [AS 702] 12 msec 8 msec 12 msec
  5 158.43.233.242 [AS 702] 12 msec 260 msec 220 msec
  6 146.188.7.226 [AS 702] 248 msec 8 msec 12 msec
  7 146.188.8.169 [AS 702] 88 msec 84 msec 88 msec
  8 146.188.13.33 [AS 702] 84 msec 84 msec 84 msec
  9 152.63.9.194 [AS 701] 84 msec 84 msec 88 msec
 10 152.63.38.70 [AS 701] 84 msec 84 msec 88 msec
 11 152.63.38.133 [AS 701] 84 msec 84 msec 84 msec
 12 204.255.173.10 [AS 701] 88 msec 88 msec 88 msec
 13 154.54.2.201 [AS 174] 88 msec 88 msec 92 msec
 14 66.28.4.209 [AS 174] 164 msec 328 msec 180 msec
 15 38.112.6.226 [AS 174] 160 msec 156 msec 156 msec
 16 137.164.22.168 [AS 2152] 160 msec 156 msec 156 msec
 17 137.164.22.111 [AS 2152] 164 msec 164 msec 204 msec
 18 137.164.32.189 [AS 2152] 164 msec 160 msec 164 msec
 19 165.196.153.26 [AS 2152] !H  !H  !H
```

Most single source IP addresses are actively seeking a response. The only obstacle which questions whether the source address is spoofed, is the fact there is no completion of the 3-way handshake. After looking at previous detects with single source IP addresses, I have found some analysts associating a medium spoofing probability. However, some based their analysis on the IP header checksum field, which is discounted as being an abnormality above. [34] Furthermore, the TCP 3-way handshake was most likely not completed because the target address does not know how to interpret the Detect#1 packet or the 2002.6.11 log file did not capture the traffic.

Attack Description

Based on the above analysis, this specific packet is not an attack. There is only one packet to analyze with this alert. If more than one packet triggered the (snort_decoder) Truncated TCP Options alert then further analysis could have been performed and may have resulted in a different outcome. Additionally, if the target host would have responded to the source host request, further analysis might have gleaned more information.

Further research of the (snort_decoder): Truncated TCP Options alert shows no direct link to any well known attack. It shows only descriptions of the alert, the alert being generated on varying IDS reporting sites, references on how to disable the alert, and one reference that the alert may be triggering on possible scan activity.

Snort decoder description.
http://www.mcabee.org/lists/snort-users/Apr-03/msg01145.html

[34] nsck2000@yahoo.com. DSHIELD: LOGS: GIAC GCIA Version 3.3 Practical Detect(s) b. 31 Aug 2002. <http://www.dshield.org/pipermail/intrusions/2002-August/005072.php>

Deleted: → - → -

Deleted: - -

Shows one recorded alert
http://www.security.org.sg/gtec/honeynet/viewdiary.php?diary=20040926

Evidence of alert capturing active port scans
http://text.dslreports.com/forum/remark,9791517

Some list this as an attack
http://www.venom600.org/code/SnortSlinger/

Inconclusive analysis of packet.
http://www.packetshack.org/index.php?page=snort_trunc_opt

Most references want to disable this alert in the snort.conf file.
http://www.linuxquestions.org/questions/archive/4/2004/02/4/150670

Through the above references, it can be implied that if repeated (snort_decoder) Truncated TCP Options alerts and responses from the targeted system, the attack would be considered a reconnaissance effort using OS fingerprinting.

Attack Mechanism

There is no attack mechanism since the above analysis has deemed this a false - positive. Past analysis on (snort_decoder) Truncated TCP Options alert has been quoted by Neil Dickey, Ph.D. “When I check my web server logs for the source IP and the time, I have so far always found that these alerts are generated during legitimate sessions.” [35] No response from the target host was captured in any of the log files. Further analysis of this packet is limited. As suggested above, repeated occurrences of this attack may signal a reconnaissance effort and a prelude to a more serious attack.

Correlations

Does the source host reply?

```
wd -nr c:\snort\bin\2002.6.11 -vv src host 46.5.180. 133 >
c:\Ryan\GCIA\Part2\Source\Does_target_reply

21:53:56.074488 IP (tos 0x0, ttl 63, id 14351, len 587) 46.5.180.133.80
> 147.91.1.45.35343: P [bad TCP cksum c96c (->e39c)!]
3574766657:3574767204(547) ack 3241705904 win 32120 (DF)bad cksum 9091
(->8a8b)!
21:53:57.154488 IP (tos 0x0, ttl 63, id 14358, len 599) 46.5.180.133.80
> 147.91.1.41.4984: P [bad TCP cksum 185b (->328b)!]
357129796:357129834 3(547) ack 3506514742 win 31856 <nop,nop,timestamp
283935 1822338> (DF)bad cksum 9082 (->8a7c)!
```

[35] Dickey, Neil Ph.D. Security Focus Incident Archive: <http://www.securityfocus.com/archive/75/319981/2003-04-24/2003-04-30/0>

Deleted: → - → -

Deleted: - -

```
21:53:57.724488 IP (tos 0x0, ttl 63, id 14365, len 599) 46.5.180.133.80
> 147.91.1.41.4991: P [bad TCP cksum 5187 (->6bb7)!]
3573807056:3573807603(547) ack 3508250192 win 31856 <nop,nop,timestamp
283993 1822446> (DF)bad cksum 907b (->8a75)!
23:29:22.744488 IP (tos 0x0, ttl 63, id 4454, len 594) 46.5.180.133.80
> 195.29.208.9.47537: P [bad TCP cksum 700e (->a028)!]
1038045389:1038045931(542) ack 968323269 win 31856 <nop,nop,timestamp
856420 231250163> (DF)bad cksum b894 (->b28e)!
23:29:25.844488 IP (tos 0x0, ttl 63, id 4461, len 594) 46.5.180.133.80
> 195.29.208.9.47614: P [bad TCP cksum 90bc (->c0d6)!]
1045041376:1045041918(542) ack 971804608 win 31856 <nop,nop,timestamp
856729 231250708> (DF)bad cksum b88d (->b287)!
09:34:52.924488 IP (tos 0x0, ttl 63, id 35535, len 576) 46.5.180.133.80
> 213.191.135.229.1381: P [bad TCP cksum 5b43 (->553d)!]
748644380:748644916(536) ack 21048548 win 32696 (DF)bad cksum 74bf (-
>6eb9)!
```

The source host does reply to port 80 traffic with a 64 byte initial TTL. This appears to be a web ser ver running on a unix based platform.

Gathering information on the source host:
To get a better comprehension of any other network traffic from the source host crossing the sensor, the below windump filters were created. I search for any packets sent fro m the source host of 165.196.153.26. I also, searched for any response to the source host of 165.196.153.26.

```
wd -r c:\snort\bin\2002.6.17 -n src host 165.196.153.26 >
c:\Ryan\GCIA\Part2\anymoresrc\anymoresrcdetect_2002.6.17  ←ran
for all raw log files.
```

```
wd -r c:\snort\bin\2002.6.18 -n dst host 165.196.153.26 >
c:\Ryan\GCIA\Part2\anymoresrc\anymoresrcdetect_2002.6.18  ←ran
for all raw log files.
```

No additional results. Only one data packet was discovered in the 2002.6.11 log file. Trying to uncover somethi ng, I searched for any additional source ports of 2161 and came up with the same results as the above queries.

```
wd -r c:\snort\bin\2002.6.18 -n src port 2161 >
c:\Ryan\GCIA\Part2\anymoresrc\anymoresrcdetect_2002.6.18  ←ran
for all raw log files.
```

No results. Additionally, D -Shield shows no known Trojans on port 2161. [36]
The raw logs show only a single packet sent from the source IP address of 165.196.153.26 destine to 46.5.180.133 target host. Next, a windump filter was created and run to determine if any pa ckets were sourced from a 165.196.0.0 subnet for all raw log files

[36] DShield. Port Report: 2161. 27 Oct 2004. <http://www.dshield.org/port_report.php?port=2161&recax=1&tarax=2&srcax=2&percent=N&days=40&Redraw=Submit+Query>

Deleted: → - → -

Deleted: - -

```
wd -nr c:\snort\bin\2002.5.4  -vv src host 165.196 or dst host
165.196 > c:\Ryan\GCIA\Part2\Source\detect_2002.5.4
```

No additional packets to analyze from the 165.196 target subnet. More information is needed for additional correlation.

Evidence of Active Targeting

There does not appear to be any active targeting. The Source host is sending a normal http get request. The packets beginning data was read into the first 8 bits reserved for TCP options for an unknown reason. IP addresses in Detect#1 have not been directly re-sanitized based on the above analysis.

Severity

Severity = (criticality + lethality) - (system countermeasures + network countermeasures)

Criticality = 4
The source is specifically targeting port 80 and searching for an active response. The web server could be running critical e-commerce services, which is a critical part of total gross revenue.

Lethality = 2
The Truncated TCP options alert has only been successful at identifying scanning using OS fingerprinting. If the attack was successful (the target responded), the source host could possibly gather information on the OS being used within the internal network.

System Countermeasures = 5
The target host does not respond to the source host's request. The target host continued to process incoming packets from additional source hosts. The target system continued to function normally. System counter measures must be high. Additionally, if the target host IP addresses was a business critical web-server, there should be a redundant or failover web server available and on the network at all times.

Network Countermeasures = 1
Network countermeasures are low because this traffic is allowed through the perimeter defense.

Calculated Severity:
(4+2) – (5+1) = 0

Defensive Recommendation

I believe the **(snort_decoder): Truncated Tcp Options** alert is worth having enabled. This alert has the capability of picking up scanning activity from OS

Deleted: → - → -

Deleted: - -

fingerprinting per the paranoid pengu in article.[37] If this alert persisted from a specific source host to multiple target hosts within the internal network my recommendation would be to block the source host using an extended access control list at the border ISP router. In the above alert t he ACL statement would look like this on a Cisco router:

```
ACL 155
deny ip 165.196.153.26 0.0.0.0 any log
permit ip any any
```

Ingress filtering would need to be applied on the outer most facing router interface. For instance:

```
Config-interface -atm1/0.255# i p access -group 155 in
```

This would deny the source host access into the 46.5.180.0 (targeted internal network) and if accompanied by a log statement would allow network security administrators to track if the source host was still attempting to access the internal network.

Alternatively, if better rules exist for capturing OS fingerprinting, it may be appropriate to disable this rule.

Multiple Choice Test Question

If TCP options are set in 12 th byte offset, the higher order nibble will have a value greater than

A. 0x30
B. 0x50
C. 0x05
D. 0x90

Answer is B.

Detect Two [BackWeb] – STOP Calling home

22,000 Cisco NIDS Alerts in 1 week. Below is the sanitized output from ArcSight console.

Detect Time	Event Name	Source Address	Source Port	Target Port	Target Address	Device Host Name
13 Jul 2004 23:06:41 PDT	External IP Detected - Port 80	MY.NET.29.85	2503	80	THERE.NET.254.9	xxxxx-cs4235

[37] Bauer, Mick. Linux Journal: Paranoid Penguin: Checking Your Work with Scanners, Part I (of II): nmap. 1 May 2001. <http://www.linuxjournal.com/article.php?sid=4561>

Deleted: → - → -

Deleted: - -

13 Jul 2004 18:07:40 PDT	External IP Detected - Port 80	MY.NET.29.85	2284	80	THERE.NET.254.10	xxxxx-cs4235
13 Jul 2004 16:57:43 PDT	External IP Detected - Port 80	MY.NET.29.85	2191	80	THERE.NET.254.9	xxxxx-cs4235
13 Jul 2004 12:52:47 PDT	External IP Detected - Port 80	MY.NET.29.85	1852	80	THERE.NET.254.9	xxxxx-cs4235
13 Jul 2004 08:07:27 PDT	External IP Detected - Port 80	MY.NET.29.85	3878	80	THERE.NET.254.9	xxxxx-cs4235
12 Jul 2004 22:44:53 PDT	External IP Detected - Port 80	MY.NET.29.85	1357	80	THERE.NET.254.11	xxxxx-cs4235
12 Jul 2004 20:47:48 PDT	External IP Detected - Port 80	MY.NET.29.85	3791	80	THERE.NET.254.9	xxxxx-cs4235
12 Jul 2004 18:46:41 PDT	External IP Detected - Port 80	MY.NET.29.85	2002	80	THERE.NET.254.9	xxxxx-cs4235

Source of Trace

Detect#2 alerts were captured by both infrastructures detailed in the Part I. Multiple Cisco NIDS devices captured the above output. Additionally, one of the source hosts was tracked down and a raw log was taken from an ethereal packet capture in promiscuous mode . Ethereal was setup to filter everything but UDP traffic to three target addresses. However, it is important to note the below capture was not the beginning of the investigation or correlation analysis. The Correlations section will explain this in thorough detail.

```
Frame 272 (88 bytes on wire, 88 bytes captured)
Ethernet II, Src: 00:03:47:a3:12:2f, Dst: 00:00:0c:07:ac:a4
    Destination: 00:00:0c:07:ac:a4 (All-HSRP-routers_a4)
    Source: 00:03:47:a3:12:2f (Intel_a3:12:2f)
    Type: IP (0x0800)
Internet Protocol, Src Addr: MY.NET.29.85 (MY.NET.29.85), Dst Addr:
THERE.NET.254.10 (THERE.NET.254.10)
User Datagram Protocol, Src Port: 9370 (9370), Dst Port: 370 (370)
    Source port: 9370 (9370)
    Destination port: 370 (370)
    Length: 54
    Checksum: 0x1421 (correct)
Data (46 bytes)

21 24 00 8a 60 67 29 0e 00 06 16 4f 00 18 62 77   !$..`g)....O..bw
```

Deleted: → - → -

Deleted: - -

```
73 75 70 64 61 74 65 30 31 2e XX XX XX XX XX XX   supdate01.XXXXXX
XX 68 2e 63 6f 6d 16 67 00 04 00 00 00 50         XX.com.g.....P
```

Detect Generation Method

The detected was generated by a custom Cisco rule that fires on the below logic:
MY.NET > External IP address Port 80

This is a Cisco based rule which fires on any network traffic not using the corporate external web proxy infrastructure. Specific Cisco IDS alerts are configured using CiscoWorks Remote console interface. [38]

Address Spoofing Probability

No. All source addresses are valid. They reside in DNS or have valid DHCP addresses assigned.

Attack Description

Backweb vulnerabilities were first discovered in 1999. However, this traffic is still active and prevalent today (September 2004).

> "* BackWeb Vulnerability: BackWeb software is included, often by default, with new computers and software, such as anti-virus software, to enable remote distribution of updates. Because of this, many people have this running and don't realize it. Due to weak authentication, it is possible for an attacker to spoof the communications between BackWeb client and server software. Depending on the client security settings, an attacker may send executable files to be run on the client machine. [39]

The CVE description is located here:

http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0395

Approximately five hundred systems within the corporate environment are targeting THERE.NET.254.9,10,11 through port 80. DNS lookups of the systems exhibiting this traffic all show client workstations. No critical IT infrastructure shows evidence of this type of network behavior. All client workstations are patched and updated with the latest virus software promptly after completing the Rapid Risk Assessment life cycle. However, client workstations are loosely regulated on what software they can install. Taking a closer look at the client workstation above a unique ladhide3.dll was found. This .dll is specifically related to a software program called BackWeb.

[38] Cisco Systems. "Maintaining Security Monitor". <http://www.cisco.com/en/US/products/sw/cscowork/ps3991/products_user_guide_chapter09186a008018d96f.html>
[39] xforce@iss.net. Fokus: ISS Alert Mailing List: ISS News Flash. 8 Feb 1999. <http://www.fokus.gmd.de/research/cc/vst/products/Security/alert/msg00071.html>

Deleted: → - → -

Deleted: - -

> "It is a small program whose purpose is to download software or other content from the Internet. It is often bundled as part of a 3rd -party software package for the purpose of automatically downloading product updates, but others have included it for the purpose of downloading unsolicited advertising or installing unwanted software (foistware). What the Backweb client downloads is entirely dependent on what the 3rd -party developer tells it to download; it does not do its own bidding. "[40]

Attack Mechanism

BackWeb is installed unknowingly by the user on their client system when they download and install software from a well known application. Some of these applications are "Bundled with products from HP (HP Pavilion), Compaq, Network Associates, Real Networks, Logitec (with their mouse drivers!), IBM, F - Secure, Western Digital Data Lifeline, Kodak digital camera sync software, Kodak Software Updater (for Kodak Easyshare digital cameras), Packard Bell ActivSurf."[41] Knowing these companies install BackWeb , "…an attacker may send false data to a BackWeb client, acting as the real BackWeb server." [42] Additionally, the BackWeb agent software can be used to send newsflashes, marketing pop ups and, full size advertisements. This software poses a security risk and has the potential to decrease employee efficiency.

The source machines appear to broadcast out some type of "Hello, do you have an updates for me?" type packets on port 80 and 370. Both TCP port 80 and UDP port 370 traffic is blocked outbound at the firewall for non -proxied traffic. These packets will leave the source host only to be blocked at the firewall. The first line of defense caught this traffic, but unnecessary UDP traffic is still sent across expensive WAN circuits.

Correlations

Detect#2 was consistently generated by a large, but unknown number of source hosts within the corporate environment. Further analysis of Detect#2 uncovered UDP traffic destined to port 370. This was seen from the first line of defense; router ACLs. The ACL logged several hits destined to 3 specific external IP addresses from multiple valid DNS \DHCP internal source hosts. A cat command was run on the router logging server to determine the number of logged entries. Below is a sample of router output.

Cisco log

```
cat /var/log/router/cisco -info.log | grep "MY.NET.29.85"
 list 166 permitted udp MY.NET.29.85(9370) -> THERE.NET.254.9(370), 1 packet
```

[40] Trustix. "Trustix™ Personal Firewall Spyware: BackWeb / BackWeb Light Client. 2004. <http://www.personalfirewall.trustix.com/spyware/backweb.html>

[41] <http://www.pestpatrol.com/pestinfo/b/backweb.asp>

[42] Leu, Matthias. "News January 1999". 12 Feb 1999. <http://www.leu.de/security/0199_e.html>

Deleted: → - → -

Deleted: - -

```
list 166 permitted TCP MY.NET.29.85(3556) -> THERE.NET.254.11(80), 1 packet
list 166 permitted TCP MY.NET.29.85(1392) -> THERE.NET.254.9(80), 1 packet
list 166 permitted udp MY.NET.29.85(9370) -> THERE.NET.254.10(370), 1 packet
```

Router ACLs cannot be relied upon as an accurate amount of traffic being sent since an entry is only logged after a certain th reshold is hit on the router. These thresholds are dependant on the router IOS version . The NIDS were detecting the port 80 traffic, but were not seeing the UDP traffic because no signatur e matched the external UDP traffic.

Gathering information on the target:

Plugging the target into D-Shield provided immediate network identification of the target addresses.

IP Address: THERE.NET.254.9

HostName: THERE.NET.254.9

DShield Profile:

Country:	🇺🇸 US
Contact E-mail:	
AS Number:	0
Total Records against IP:	not processed
Number of targets:	select update below
Date Range:	to

request contact update

Update Summary

Whois:

```
OrgName:    XXXXXXXXX
OrgID:      XXX
Address:    XXXX XXXXXX Drive
City:       XXXXX
StateProv:  CA
PostalCode: XXXXX
Country:    US
```

<Ryan> Do you have a XXXXX Mouse?

<Mike> Yes, I purchased a wireless mouse.

<Ryan> Perfect. You have been very helpful! You can continue to use your wireless mouse.

This actual conversation transpired thru an e-mail chain Wednesday, July 14, 2004 while investigating alerts from 2 different sources. Additionally, plugging in the site IP address THERE.NET.254.9 into a web browser resulted in the below site redirection.

http://THERE.NET .254.9

Deleted: → - → -

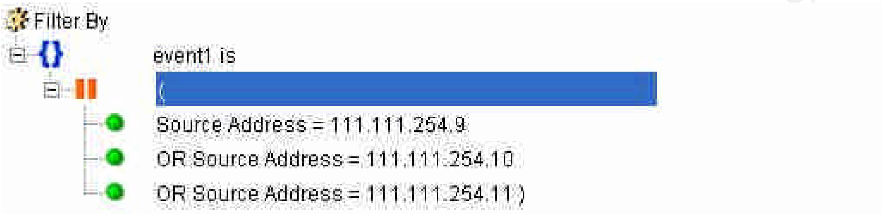
Deleted: - -

```
You are accessing a BackWeb channel (it is not a norm  al Web
site).

To learn about BackWeb, click  here.

<end>
```

When being redirected or clicking the link above the user is redirected to http://www.personalfirewall.trustix.com/spyware/backweb.html

At this stage, some network characterization has been completed. The source and cause of the network traffic has been identified. Based on the target address and associated company the packets do no t seem to be malicious. Further analysis on this packet could be done if the traffic was allowed outside the firewall.

Lets check ArcSight for any responses from a THERE.NET.254.9,10,11. THERE.NET is substituted for 111.111 because ArcSight will not all ow non IP addresses in the Zone editor shown below.



```
Filter By
  {}        event1 is
    ||      (
      ●     Source Address = 111.111.254.9
      ●     OR Source Address = 111.111.254.10
      ●     OR Source Address = 111.111.254.11 )
```

Arcsight returns no results. This confir ms the firewall is blocking outside response from the target hosts . This also confirms that no spoofed hosts have responded inside of the internal WAN infrastru cture.

Evidence of Active Targeting

Every source who has installed XXXXX wireless mouse has the BackWeb agent installed. Approximately 500 source host in the corporation.

Severity

Severity = (criticality + lethality) - (system countermeasures + network countermeasures)

Criticality = 2
The source is a single workstation or laptop not running any critical IT infrastructure. These devices may contain sensitive proprietary information.

Lethality = 2
These are only client workstations. Cannot find any expli cit or destructive exploits, only discussion of spyware and adaware components. [43]

Deleted: → - → -

Deleted: - -

[43] Mason, Dave. Segment 7. 5 Jun 2004. <http://www.davemason.com/june04.html>

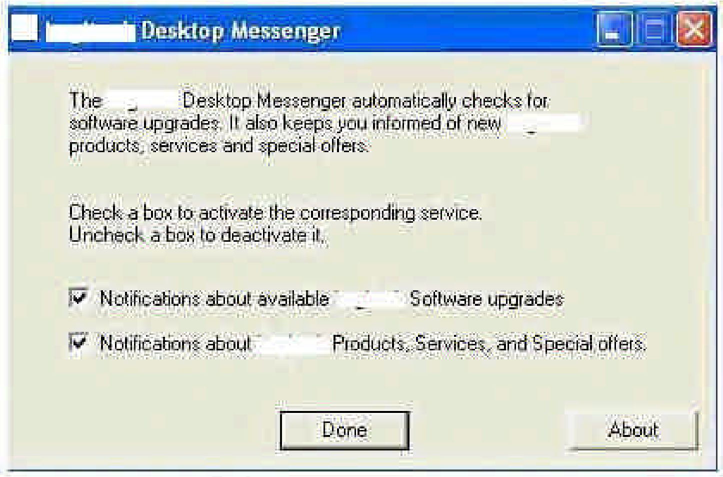
System Countermeasures = 0
The source host sends requests to target hosts. The system counter measures are low. The system is currently defenseless. Installed patches and virus software currently do not posses capabilities to remove this type of SpyWare.

Network Countermeasures = 5
Network countermeasures are high because this traffic is not allowed through the perimeter defense.

Calculated Severity:
(2+2) – (5+0) = -1

Defensive Recommendation

There are some recommendations on r emoving BackWeb.[44] However, for a corporate environment getting all 500 users to run through the recommended removal process will consume a large amount of corporate resources. The UDP and TCP traffic can actually be eliminated by deselecting options listed in the software application. This method is actually recommended by Dave Mason in his 6/5/04 segment 7 audio.[45] Sending an e-mail to these 500 employees instructing them to deselect the below options is a viable recommendation.



Desktop Messenger

The Desktop Messenger automatically checks for software upgrades. It also keeps you informed of new products, services and special offers.

Check a box to activate the corresponding service.
Uncheck a box to deactivate it.

☑ Notifications about available Software upgrades

☑ Notifications about Products, Services, and Special offers.

[44] "BackWeb Removal
How to remove and uninstall backweb adware". <http://www.spysweeper.com/backweb-removal.html>
[45] Mason, Dave. Segment 7. 5 Jun 2004. <http://www.davemason.com/june04.html>

Deleted: → - → -

Deleted: - -

Multiple Choice Test Question

Which company is not known for integrating BackWeb into one or more of their products?

A. Kodak
B. Logitec
C. Mobile Excitement
D. HP

Answer is C.

Detect Three [Unauthorized Scan]

ArcSight alert console reports:
~500 SNMP Violations
~900 TCP SYN Host Sweep
~5,000 Net Sweep -Echo

Source of Trace

Below is a sample of the output to the ArcSight Alert console.

Event Name	Source Address	Source Port	Target Address	Target Port
SNMP Violation	MY.NET.42.107	62768	THERE.NET.113.92	161
SNMP Violation	MY.NET.42.107	62768	THERE.NET.113.92	161
SNMP Violation	MY.NET.42.107	62705	THERE.NET.113.92	161
SNMP Violation	MY.NET.42.107	62705	THERE.NET.113.92	161
SNMP Violation	MY.NET.42.107	62635	THERE.NET.113.88	161
SNMP Violation	MY.NET.42.107	62635	THERE.NET.113.88	161
SNMP Violation	MY.NET.42.107	62635	THERE.NET.113.88	161
SNMP Violation	MY.NET.42.107	62635	THERE.NET.113.88	161
SNMP Violation	MY.NET.42.107	62582	THERE.NET.113.88	161
SNMP Violation	MY.NET.42.107	62582	THERE.NET.113.88	161
SNMP Violation	MY.NET.42.107	62572	THERE.NET.113.90	161
SNMP Violation	MY.NET.42.107	62572	THERE.NET.113.90	161
SNMP Violation	MY.NET.42.107	62567	THERE.NET.113.91	161
SNMP Violation	MY.NET.42.107	62582	THERE.NET.113.88	161
SNMP Violation	MY.NET.42.107	59343	THERE.NET.113.54	161
SNMP Violation	MY.NET.42.107	59279	THERE.NET.113.53	161
SNMP Violation	MY.NET.42.107	59279	THERE.NET.113.53	161
SNMP Violation	MY.NET.42.107	59244	THERE.NET.113.51	161
SNMP Violation	MY.NET.42.107	59244	THERE.NET.113.51	161
SNMP Violation	MY.NET.42.107	59200	THERE.NET.113.51	161
SNMP Violation	MY.NET.42.107	59160	THERE.NET.113.32	161
SNMP Violation	MY.NET.42.107	59200	THERE.NET.113.51	161
SNMP Violation	MY.NET.42.107	59160	THERE.NET.113.32	161
SNMP Violation	MY.NET.42.107	59111	THERE.NET.113.32	161
SNMP Violation	MY.NET.42.107	59096	THERE.NET.113.30	161
SNMP Violation	MY.NET.42.107	59096	THERE.NET.113.30	161
SNMP Violation	MY.NET.42.107	59111	THERE.NET.113.32	161
SNMP Violation	MY.NET.42.107	59096	THERE.NET.113.30	161

Deleted: → - → -

Deleted: - -

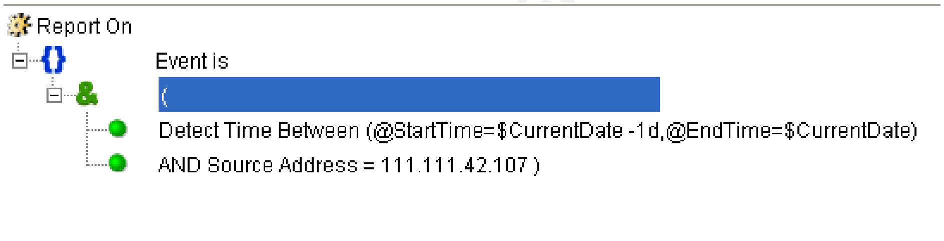
SNMP Violation	MY.NET.42.107	59096	THERE.NET.113.30	161
SNMP Violation	MY.NET.42.107	59021	THERE.NET.113.30	161
SNMP Violation	MY.NET.42.107	59021	THERE.NET.113.30	161
SNMP Violation	MY.NET.42.107	59021	THERE.NET.113.30	161
SNMP Violation	MY.NET.42.107	59021	THERE.NET.113.30	161

Combined with…

Event Name	Source Address	Target Address	Target Port	Source Port
TCP SYN Host Sweep	MY.NET.42.107	THERE.NET.113.132	1311	9779
TCP SYN Host Sweep	MY.NET.42.107	THERE.NET.113.210	1311	9779
TCP SYN Host Sweep	MY.NET.42.107	THERE.NET.113.200	1311	9779
TCP SYN Host Sweep	MY.NET.42.107	THERE.NET.113.134	411	9776
TCP SYN Host Sweep	MY.NET.42.107	THERE.NET.113.134	1311	9779
Net Sweep-Echo	MY.NET.42.107	THERE.NET.113.150	0	8
TCP SYN Host Sweep	MY.NET.42.107	THERE.NET.113.220	280	9778
TCP SYN Host Sweep	MY.NET.42.107	THERE.NET.113.149	280	9778
Net Sweep-Echo	MY.NET.42.107	THERE.NET.113.132	0	8
Net Sweep-Echo	MY.NET.42.107	THERE.NET.113.220	0	8
Net Sweep-Echo	MY.NET.42.107	THERE.NET.113.149	0	8
TCP SYN Host Sweep	MY.NET.42.107	THERE.NET.113.141	280	9778
TCP SYN Host Sweep	MY.NET.42.107	THERE.NET.113.143	280	9778
Net Sweep-Echo	MY.NET.42.107	THERE.NET.113.214	0	8
TCP SYN Host Sweep	MY.NET.42.107	THERE.NET.113.220	8008	9780

Detect Generation Method

Detect#3 alerts were captured by a single Cisco NIDS device in a Tier II infrastructure detailed in the Part I. The output is displayed as an exported ArcSight Excel.csv report with the below conditions applied. All of the Query Options selected below ar e not shown in the above output.



Report On

Event is

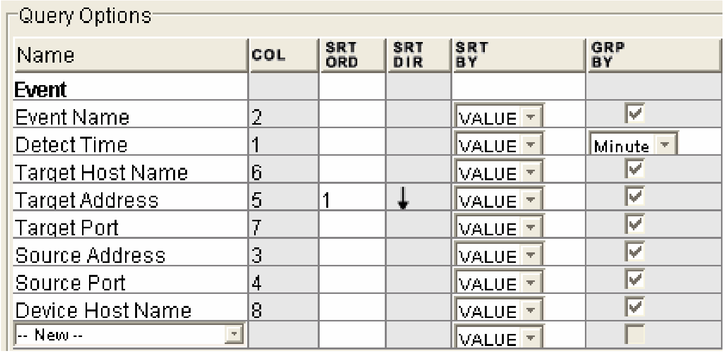
(

Detect Time Between (@StartTime=$CurrentDate -1d,@EndTime=$CurrentDate)

AND Source Address = 111.111.42.107)

Deleted: → - → -

Deleted: - -



Name	COL	SRT ORD	SRT DIR	SRT BY	GRP BY
Event					
Event Name	2			VALUE	☑
Detect Time	1			VALUE	Minute
Target Host Name	6			VALUE	☑
Target Address	5	1	↓	VALUE	☑
Target Port	7			VALUE	☑
Source Address	3			VALUE	☑
Source Port	4			VALUE	☑
Device Host Name	8			VALUE	☑
-- New --				VALUE	□

Address Spoofing Probability

No. The source address and target addresses are valid. They reside in DNS or have valid DHCP addresses assigned. The source address is assigned to a mobile laptop user→ XXXXXXX-mobl2.XXX.corp.XXXX.com. The employee of the system can be looked up on an internal directory based on the mobile DHCP address. It is possible the users address is being spoofed, but unlikely.

Attack Description

The attacks below all have detailed attack descriptions and have been previously analyzed extensively. Below will attack descriptions will only reference the CVE numbers required by the GCIA Version 3.5 requirements. The alerts by themselves are not very significant, but together show a scanning activity.

SNMP Violation:

SNMP vulnerabilities are not new. A quick search of Google for "snmp vulnerability" returns almost four thousand results. Including "The Common Vulnerabilities and Exposu res project (cve.mitre.org) has assigned the names CAN-2002-0012 and CAN -2002-0013 to these issues."[46] SNMP v1 or v2 are reserved in a corporate environment for approved network management servers due to security limitations in the protocol. Public SNMP community should be replaced by strings with strong passwords.

Net Sweep Echo :

Ever used ping? Many network management tools utilize ICMP protocol as a method of sending keep -alive requests to remote hosts.

[46] Red Hat. Errata: Updated ucd-snmp packages available. 12 Mar 2002. <http://rhn.redhat.com/errata/RHSA-2001-163.html>

Deleted: → - → -

Deleted: - -

CAN-1999-0523[47] and CAN-1999-0635[48]

TCP SYN Host Sweep:

> **3030 TCP SYN Host Sweep** —This triggers when a large number of ICMP Echo Replies are targeted at a machine. They can be from one or many sources. This will catch the attack known as Smurf, described in the related vulnerability page. Because this attack can come from many sources, automatic shunning of individual hosts is not very effective. If only one network is being used to broadcast the re plies, the network can be shunned. [49]

CVE-2000-0324[50] and CAN-1999-1373[51]

Attack Mechanism

Together, these three alerts could be a reconnaissance effort and a prelude to a more serious attack. Corporate policy tightly controls scanning of the internal network environment and limits these activities to network management tools. If there is a need for this type of activity, the SOC should be notified.

Correlations

Source & Target:

Source = MY.NET.42.86

Target = THERE.NET .113.xxx subnet

DShield References:

Port = 280

TCP	http-mgmt	
Udp	http-mgmt	

Port = 411

TCP	Backage	[trojan] Backage
TCP	Rmt	Remote MT Protocol
udp	Rmt	Remote MT Protocol

[47] Common Vulnerabilities and Exposures. CAN-1999-0523. 26 Jul 1999. <http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0523>

[48] Common Vulnerabilities and Exposures. CAN-1999 -0635. 8 Aug 1999. <http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0635>

[49] Cisco Systems. "Cisco Secure Intrusion Detection System Version 2.2.0 Release Notes". 1998. <http://www.cisco.com/en/US/products/sw/secursw/ps5052/prod_release_note09186a00800ee999.html>

[50] Common Vulnerabilities and Exposures. CVE-2000-0324. 9 Mar 2002. <http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2000-0324>

[51] Common Vulnerabilities and Exposures. CAN-1999-1373. 12 Sept 2001. <http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-1373>

Deleted: → - → -

Deleted: - -

Port = 8008

TCP	http-alt	HTTP Alternate
TCP	novell-http	Novell Netware Management Protocol
udp	http-alt	HTTP Alternate

Based on the above alerts and correlation the source device is illustrating symptoms of managing devices on the THERE.NET.113.xxx subnet thru ICMP and SNMP through a personal laptop. The employee is either running a mis-configured device or unaware of the company's security policies and guidelines. The employee could be running some type of network management utility from his\her laptop.

The SOC should be the single point of contact for employees inquiring about executing scans.

Evidence of Active Targeting

This traffic is directed at the THERE.NET.113.xxx subnet only.

Severity

Severity = (criticality + lethality) - (system countermeasures + network countermeasures)

Criticality = 3
The source is a single laptop. The targeted subnet contains critical IT infrastructure, which would impact production if affected. This could be indicative of a future attack.

Lethality = 2
The current alerts are only scanning activity.

System Countermeasures = 2
The source hosts are responding. The system counter measures are low. The system are currently giving up potential information about there OS, patch level, ect.

Network Countermeasures = 0
Network countermeasures are low because this there is no perimeter defense.

Calculated Severity:
(3+2) – (2+0) = 3

Deleted: → - → -

Deleted: - -

Defensive Recommendation

Employee needs to justify business need to perform scanning activities. If the employee cannot justify there scanning activity, the user needs to discontinue their scanning\SNMP\ICMP activities.

An E-mail to the user should be sent immediately:

It appears you ran some type of network management utility from your laptop which included Scanning, ICMP and SNMP requests. These activities started May 27th @ 2004-05-27-15:03 PST and May 28th at 12:39:21PST.

Are you aware of this activity being run from your laptop? Can you name of the application?

If you have a business justification for performing this activity please be aware of the company policy regarding both Scanning and SNMP management.
Guidelines can be found on http://secure.XXXXX.com/InfoSec
Network Security Center

Multiple Choice Test Question

Scanning across a network should be limited to
A. Anyone
B. 10 Employees
C. SOC or designated personal
D. Terminated Employees

Answer is C.

Part Three Analyze This

Executive Summary

There are areas below listed in order of importance.

1. Several compromised and potentially compromised internal hosts are listed in the below report. Internal hosts need to be patched and cleaned with up-to-date software. This seems like a broken record response, so a longer term solution is to implement a point-of-entry solution which checks for the latest patches/virus software before internal hosts are allowed to connect to the network. Implementing a point-of-entry solution will clean up the network and eliminate many alerts.
2. Stricter firewall rules need to be established blocking the below ports:
   69
   135
   137

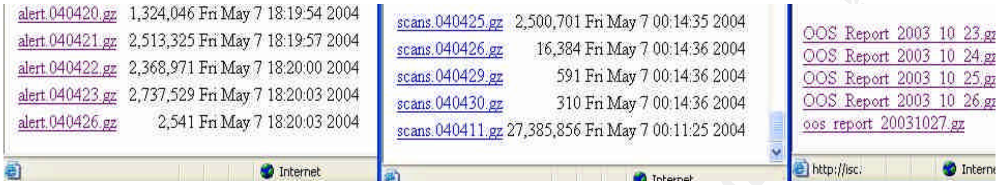
Deleted: → - → -

Deleted: - -

138
139
445
5900

3. Extensive filtering needs to be done to the eliminate false positives. Thousands of alerts are being generated by legitimate network traffic.
4. A brief summary of the university’s acceptable net work usage policy or a written document establishing network usage guidelines for users would greatly focus attention and efforts of future Intrusion Analysis.
5. Lastly, the university should contact their ISP to see if any additional defensive measures can be made to reduce the external scanning and exploit attempts on the university network address space.

Files Analyzed



```
alert.040420.gz  1,324,046 Fri May 7 18:19:54 2004
alert.040421.gz  2,513,325 Fri May 7 18:19:57 2004
alert.040422.gz  2,368,971 Fri May 7 18:20:00 2004
alert.040423.gz  2,737,529 Fri May 7 18:20:03 2004
alert.040426.gz      2,541 Fri May 7 18:20:03 2004

scans.040425.gz  2,500,701 Fri May 7 00:14:35 2004
scans.040426.gz     16,384 Fri May 7 00:14:36 2004
scans.040429.gz        591 Fri May 7 00:14:36 2004
scans.040430.gz        310 Fri May 7 00:14:36 2004
scans.040411.gz 27,385,856 Fri May 7 00:11:25 2004

OOS_Report_2003_10_23.gz
OOS_Report_2003_10_24.gz
OOS_Report_2003_10_25.gz
OOS_Report_2003_10_26.gz
oos_report_20031027.gz
```

The last 5 logs as stated in the GCIA versio n 3.5 practical assignment guid eline were analyzed . Note: The log names do not match up fo r OOS files. To analyze the log files, Tu Niem’s practical was referenced and the Description Analysis Process was utilized.[52] This details a 3 page description on how the above log files were manipulated and processed through SnortSnarf.

Sequence of cmd s used:

```
cat alert.04042[0 -6] > alerts

cat alert.04042[0 -6] | wc -l
 796920

wc -l alerts
 796920 alerts

grep "192.181" alerts
```

Note: grepping 192.180 returned output res ults and could not be used to replace MY.NET

```
grep -v "spp_portscan" alerts > alertsf inal2
```

[52] Niem, Tu. “Intrusion Detection in Depth”. Pgs 68-70. 23 Jan 2003. <http://www.giac.org/practical/GCIA/Tu_Niem_GCIA.pdf>

Deleted: → - → -

Deleted: - -

Defensive recommendations and the analysis process \cmds are integrated within the top 20 source and top 20 destination analysis sections. Loading alerts based off of Source Alert#3 into ArcSight and displaying them in a link graph would have brought continuity to the whole paper. However, due to unavailable access to an ArcSight test lab the link graph has been omitted.

Top Talkers

Top 20 Source IPs from SnortSnarf Output.
This page provides summary information about alerts acquired using input module SnortFileInput, wit h sources:

- alertsfinal2

The most active source IPs are shown. Rank is determined by the number of alerts with that IP as the source. Within a rank, IPs are sorted by # of signatures, then by IP number.

Rank	Total # Alerts	**Source IP**	# Signatures triggered	Destinations involved
Source #1	21788 alerts	**134.192.42.11**	10 signatures	192.181.30.4
Source #2	5206 alerts	**131.92.177.18**	1 signatures	192.181.30.3
Source #3	4768 alerts	**209.164.32.205**	9 signatures	(13 destination IPs)
Source #4	3730 alerts	**68.55.155.26**	1 signatures	192.181.30.4
Source #5	3470 alerts	**69.136.228.63**	1 signatures	192.181.30.4
Source #6	3230 alerts	**192.181.43.8**	1 signatures	(7 destination IPs)
Source #7	3109 alerts	**192.181.11.4**	2 signatures	(54 destination IPs)
Source #8	3073 alerts	**64.12.24.34**	1 signatures	(3 destination IPs)
Source #9	2990 alerts	**192.181.69.232**	2 signatures	67.167.20.228, 67.167.3.240
Source #10	2611 alerts	**220.197.192.39**	3 signatures	(181 destination IPs)
Source #11	2509 alerts	**192.181.11.7**	2 signatures	(3 destination IPs)
Source #12	2478 alerts	**69.138.77.62**	2 signatures	192.181.30.4, 192.181.30.3
Source #13	2454 alerts	**151.196.115.104**	1 signatures	192.181.30.3
Source #14	2331 alerts	**64.12.24.35**	3 signatures	(3 destination IPs)
Source #15	2191 alerts	**68.34.94.70**	2 signatures	192.181.30.4, 192.181.30.3

Deleted: → - → -

Deleted: - -

Source #16	2123 alerts	192.181.43.13	1 signatures	(3 destination IPs)
Source #17	1598 alerts	68.33.49.146	1 signatures	192.181.30.4
Source #18	1515 alerts	67.167.3.240	1 signatures	192.181.69.232
Source #19	1348 alerts	195.36.245.141	1 signatures	192.181.153.81
Source #20	1155 alerts	24.43.50.166	4 signatures	(13 destination IPs)

Source Alert#1 - *134.192.42.11*

The external source host (134.192.42.11) targets the below internal host (192.181.30.4) creating thousands of alerts. SnortSnarf reports 21779 instances of 192.181.30.4 activity from this external host. One output is shown below.

```
[**] 192.181.30.4 activity [**] 134.192.42.11:61714 -
>192.181.30.4:51443
```

See Destination Alert#1 for description on target port 51443. Other Practicals have already addressed this alert. The above alert should be filtered. What is left once target port 51443 is filtered?

```
cat alertsfinal2 | grep "134.192.42.11" | egrep  -v ":51443" |
more
```

```
[**] 192.181.30.4 activity [**] 134.192.42.1104/22-18:53:17.357969
[**] Tiny Fragments - Possible Hostile Activity [**] 209.164.32.205 ->
192.181.97.55
[**] 192.181.30.4 activity [**] 134.192.42.1104/22-18:53:40.426783
[**] Tiny Fragments - Possible Hostile Activity [**] 209.164.32.205 ->
192.181.97.55
[**] 192.181.30.4 activity [**] 134.192.42.1104/22-18:44:57.165574
[**] Null scan! [**] 61.48.8.56:62975 -> 192.181.112.209:49524
[**] 192.181.30.4 activity [**] 134.192.42.1104/22-19:06:09.747384
[**] SMB Name Wildcard [**] 192.181.11.7:137 -> 169.254.0.0:137
[**] 192.181.30.4 activity [**] 134.192.42.11:62185 ->
192.181.30.404/22-18:58:04.435305
[**] Tiny Fragments - Possible Hostile Activity [**] 209.164.32.205 ->
192.181.97.55
[**] 192.181.30.4 activity [**] 134.192.42.11:62190 ->
192.181.30.404/22-18:58:56.218007
[**] Tiny Fragments - Possible Hostile Activity [**] 209.164.32.205 ->
192.181.97.55
```

~99% of the alerts are filtered by the above grep. Interesting traffic remaining shows 209.164.32.205, this is addressed by source alert#3.

Source Alert#2 - *131.92.177.18*

```
[**] 192.181.30.3 activity [**] 131.92.177.18:1033 -> 192.181.30.3:524
```

Deleted: → - → -

Deleted: - -

Output shows target port 524, which is Network Time Protocol and the above alert triggers every 7 -9 minutes. Source Host appears to be legitimate. This is the only alert generated by this host. Filter this alert for this host.

IP Address: 131.92.177.18

HostName: aeclt-cf00a4.apgea.army.mil

DShield Profile:

Country:	🇺🇸 US
Contact E-mail:	AMOS@APGEA .ARMY.MIL
AS Number:	213

Source Alert#3 - *209.164.32.205*

Source Null scanned the university network.

```
[**] Null scan! [**] 209.164.32.205:0 -> 192.181.81.116:0
[**] Probable NMAP fingerprint attempt [**] 209.164.32.205:0 ->
192.181.81.116:0
[**] Null scan! [**] 209.164.32.205:44 -> 192.181.97.21:64760
```

TCP	Arctic	[trojan] Arctic
TCP	mpm-flags	MPM FLAGS Protocol
udp	mpm-flags	MPM FLAGS Protocol

Defensive recommendations have been suggested to block fragments at the firewall.[53] Additionally, this host has been captured scanning multiple times .[54]

IP Address: 209.164.32.205

HostName: 209.164.32.205.ptr.us.xo.net

DShield Profile:

Country:	🇺🇸 US
Contact E-mail:	abuse@xo.com
AS Number:	2828
Total Records against IP:	213
Number of targets:	3
Date Range:	2004-08-17 to 2004-11-01

Update Summary

Last Fightback Sent: not sent

[53] Breault, Steve. "SANS Intrusion Detection in Depth". 2004. <http://www.giac.org/practical/GCIA/Steve_Breault_GCIA.pdf>

[54] 2 Oct 2004. <http://netflow3.nhltc.edu.tw/netflow/scan/c6509/2004/2004-10/2004-10-02/2004-10-02.202131-OUT.html>

Deleted: → - → -

Deleted: - -

Whois:

```
OrgName:    XO Communications
OrgID:      XOXO
Address:    Corporate Headquarters
Address:    11111 Sunset Hills Road
City:       Reston
StateProv:  VA
PostalCode: 20190-5339
Country:    US

ReferralServer: rwhois://rwhois.eng.xo.com:4321/

NetRange:   209.164.0.0 - 209.164.63.255
CIDR:       209.164.0.0/18
NetName:    XOXO-BLK-18
NetHandle:  NET-209-164-0-0-1
Parent:     NET-209-0-0-0-0
NetType:    Direct Allocation
NameServer: NAMESERVER.CONCENTRIC.NET
NameServer: NAMESERVER1.CONCENTRIC.NET
NameServer: NAMESERVER2.CONCENTRIC.NET
NameServer: NAMESERVER3.CONCENTRIC.NET
Comment:    For best results, please send all spam and
worm reports only to abuse@xo.com.
RegDate:    1997-11-14
Updated:    2003-08-08

OrgAbuseHandle: XCNV-ARIN
OrgAbuseName:   XO Communications, Network Violations
OrgAbusePhone:  +1-866-285-6208
OrgAbuseEmail:  abuse@xo.com

OrgTechHandle: XCIA-ARIN
OrgTechName:   XO Communications, IP Administrator
OrgTechPhone:  +1-703-547-2000
OrgTechEmail:  ipadmin@eng.xo.com

# ARIN WHOIS database, last updated 2004-11-04 19:10
# Enter ? for additional hints on searching ARIN's WHOIS
database.


OrgName:    American Registry for Internet Numbers
OrgID:      ARIN
Address:    3635 Concorde Parkway, Suite 200
City:       Chantilly
StateProv:  VA
PostalCode: 20151
Country:    US

NetRange:   209.0.0.0 - 209.255.255.255
CIDR:       209.0.0.0/8
NetName:    NET209
NetHandle:  NET-209-0-0-0-0
Parent:
NetType:    Allocated to ARIN
```

Deleted: → - → -

Deleted: - -

```
NameServer: chia.arin.net
NameServer: dill.arin.net
NameServer: epazote.arin.net
NameServer: figwort.arin.net
NameServer: BASIL.ARIN.NET
NameServer: henna.arin.net
NameServer: indigo.arin.net
Comment:    Formerly delegated to the InterNIC
RegDate:    1996-06-01
Updated:    2004-07-22

OrgNOCHandle: ARINN-ARIN
OrgNOCName:   ARIN NOC
OrgNOCPhone:  +1-703-227-9840
OrgNOCEmail:  noc@arin.net

OrgTechHandle: ARIN-HOSTMASTER
OrgTechName:   Registration Services Department
OrgTechPhone:  +1-703-227-0660
OrgTechEmail:  hostmaster@arin.net

# ARIN WHOIS database, last updated 2004-11-04 19:10
# Enter ? for additional hints on searching ARIN's WHOIS
database.


OrgName:    XO Communications
OrgID:      XOXO
Address:    Corporate Headquarters
Address:    11111 Sunset Hills Road
City:       Reston
StateProv:  VA
PostalCode: 20190-5339
Country:    US
Comment:
RegDate:
Updated:    2003-12-16

ReferralServer: rwhois://rwhois.eng.xo.com:4321/

AbuseHandle: XCNV-ARIN
AbuseName:   XO Communications, Network Violations
AbusePhone:  +1-866-285-6208
AbuseEmail:  abuse@xo.com

AdminHandle: XCIA-ARIN
AdminName:   XO Communications, IP Administrator
AdminPhone:  +1-703-547-2000
AdminEmail:  ipadmin@eng.xo.com

TechHandle: XCIA-ARIN
TechName:   XO Communications, IP Administrator
TechPhone:  +1-703-547-2000
TechEmail:  ipadmin@eng.xo.com

# ARIN WHOIS database, last updated 2004-11-04 19:10
```

Deleted: → - → -

Deleted: - -

```
# Enter ? for additional hints on searching ARIN's WHOIS
database.
```

Source Alert#4 → 68.55.155.26 ← Elkridge, MD

```
[**] 192.181.30.4 activity [**] 68.55.155.26:1257 -> 192.181.30.4:8009
[**] 192.181.30.4 activity [**] 68.55.155.26:1257 -> 192.181.30.4:8009
```

Target port of 8009 is the only conclusive evidence. Source ports are all non-unique ephemeral. Port 8009 relates to Netware.

TCP	Netware-rmgr	Novell Netware Remote Manager

```
cat alertsfinal2 | grep "68.55.155.26" | egrep  -v ":8009" |more
```

No results. This should be filtered. Previous Practicals mention universities using Novel Netware.[55] Additionally, these two external IP addresses seem to be utilizing the NetWare service heavily.
68.55.155.26 → only activity alerts to port 8009
68.55.158.146 → extra activity to port 80 appears normal .

```
[**] 192.181.30.3 activity [**] 68.55.158.146:60174 -> 192.181.30.3:80
```

Source Alert#5 → *69.136.228.63*

```
[**] 192.181.30.4 activity [**] 69.136.228.63:3156 ->
192.181.30.4:51443
```

```
cat alertsfinal2 | grep "69.136.228.63" | egrep  -v ":51443" |
more
```

No results. Source is legitimate DSL user f ilter. This user should be filtered.

Source Alert#6 → *192.181.43.8*

This source alert is addressed by Destination Alert#3.

Source Alert#7 → *192.181.11.4*

```
[**] SMB Name Wildcard [**] 192.181.11.4:137 -> 210.120.128.117:137
```

A full investigation has already bee n performed in previous Practicals regarding this alert.[56]

TCP	Netbios-	NETBIOS Name

[55] Kroeger, Tim. "Security Information Management Systems: What are they? Who makes them? Do I need one?". 18 May 2004. <http://www.giac.org/practical/GCIA/Tim_Kroeger_GCIA.pdf>

[56] Breault, Steve. "SANS Intrusion Detection in Depth". 2004. <http://www.giac.org/practical/GCIA/Steve_Breault_GCIA.pdf>

Deleted: → - → -

Deleted: - -

	ns	Service
udp	Netbios-ns	NETBIOS Name Service
TCP	Chode	[Trojan] Chode
TCP	Qaz	[Trojan] Qaz
udp	Msinit	[Trojan] Msinit

Source Alert#8 and Destination Alert #4→ *64.12.24.34*

```
[**] High port 65535 TCP - possible Red Worm - traffic [**]
64.12.24.34:65535 -> 192.181.43.13:1605
[**] High port 65535 TCP - possible Red Worm - traffic [**]
192.181.43.13:1605 -> 64.12.24.34:65535
```

Port 1605
The three internal IP addresses below a ppear to have salutation manager installed.[57] If policy allows this application in the environment, the above alert should be filtered for target port 1605.

```
cat alertsfinal2 | grep "64.12.24.34" | egrep -v
"192.181.43.4|192.181.43.13|192.181.43.8" | more
```

Source Alert#9 → *192.181.69.232*

```
[**] High port 65535 TCP - possible Red Worm - traffic [**]
67.167.3.240:65535 -> 192.181.69.232:2894
[**] High port 65535 TCP - possible Red Worm - traffic [**]
192.181.69.232:2894 -> 67.167.3.240:65535
```

Port 2894

TCP	Abacus-remote	ABACUS-REMOTE
udp	Abacus-remote	ABACUS-REMOTE

An energy cost savings effort and \or continuous monitoring of the Heating \AC unit at this university could be underway. This is not a virus. The source application should be v erified and filtered from the IDS. All traffic is between the above two IP addresses and triggering the above alerts.[58] The below output is from the Abacus web site.

> Performance contracts for higher education facilities can include services such as:
>
> - Complete energy audits to identify energy conservation measures

[57] The Salutation Consortium. "Salutation Personalities". 2004. <http://www.salutation.org/techtalk/person.htm>

[58] Abacus. "Performance Assurance". 2004. <http://www.abacus-engr.com/services/performance_assurance/remote_diagnostics.html>

Deleted: → - → -

Deleted: - -

- Engineering design for recommended measures
- Construction/implementation of measures
- Construction management; testing and operation training
- Performance monitoring and verification. [59]

Running the below grep eliminates 99% of the alerts. This alert should be filtered after the above network characterization is confirmed.

```
cat alertsfinal2 | grep "192.181.69.232" | egrep   -v ":2894" |more
```

Source Alert#10 → *220.197.192.39*

HostName: 220.197.192.39

DShield Profile:

Country:	🇨🇳 CN
Contact E-mail:	ip_address@cnuninet.com
AS Number:	9800

```
% [whois.apnic.net node-2]
% Whois data copyright terms
http://www.apnic.net/db/dbcopyright.html
inetnum:       220.192.0.0 - 220.207.255.255
netname:       UNICOM
descr:         China United Telecommunications Corporation
descr:         No.133,Taiyun Building,Xidan North Street
descr:         Xicheng District,Beijing,China
country:       CN
admin-c:       UCH1-AP
tech-c:        UC6-AP
mnt-by:        MAINT-CNNIC-AP
mnt-lower:     MAINT-CN-CNNIC-UNICOM
changed:       hm-changed@apnic.net 20021211
status:        ALLOCATED PORTABLE
source:        APNIC
role:          Unicom China Hostmaster
address:       911 Room,Xin Tong Center,No.8 Beijing Railway Station
address:       East Avenue, Beijing,PRC.
country:       CN
phone:         +86-10-6527-8866
fax-no:        +86-10-6526-0124
e-mail:        ip_address@cnuninet.com
admin-c:       RX9-AP
tech-c:        RX9-AP
nic-hdl:       UCH1-AP
notify:        ip_address@cnuninet.com
mnt-by:        MAINT-CN-CNNIC-UNICOM
changed:       hostmaster@apnic.net 20010820
source:        APNIC
person:        Unicom China
address:       911 Room,Xin Tong Center,No.8 Beijing Railway Station
```

[59] Abacus. “Higher Education Facilities”. <http://www.abacus-engr.com/portfolio/education2/index-2.html>

Deleted: → - → -

Deleted: - -

```
address:       East Avenue, Beijing,PRC.
country:       CN
phone:         +86-10-6527-8866
fax-no:        +86-10-6526-0124
e-mail:        ip_address@cnuninet.com
nic-hdl:       UC6-AP
mnt-by:        MAINT-CNNIC-AP
changed:       ip_address@cnuninet.com 20010521
changed:       hostmaster@apnic.net 20010820
source:        APNIC

cat alertsfinal2 | grep "220.197.192.39" | egrep   -v ":80" |more

[**] SMB Name Wildcard [**] 192.181.150.44:1058 -> 220.197.192.39:137
[**] SMB Name Wildcard [**] 192.181.150.198:137 -> 220.197.192.39:137
[**] SMB Name Wildcard [**] 192.181.150.198:1109 -> 220.197.192.39:137
[**] SMB Name Wildcard [**] 192.181.150.44:137 -> 220.197.192.39:137
[**] 192.181.30.3 activity [**] 220.197.192.39:24489 -
>192.181.30.3:2745
[**] Possible trojan server activity [**] 220.197.192.39:27374 ->
192.181.18.206:6129
```

Port 6129

TCP	dameware	Dameware Remote Admin

About DameWare.[60] DameWare Vulnerability.[61]

Port 2745

TCP	Bagle.C	Bagle Virus Backdoor
TCP	urbisnet	URBISNET
udp	urbisnet	URBISNET

Port 2745 is associated with the Bagel Virus. This source host (220.197.192.39) is attempting multiple exploits. This IP addresses has been scanning multiple networks in the past.[62] Additionally, this IP is recommended by others to be blocked at the perimeter firewall.[63] If there is no association with this Chinese originating IP address this IP address could be blocked. However, this is not really the best course of action. If the external IP address is blocked they can always get a new IP address in order to circumvent this first line of defense. Better precautionary measures would include making sure the environment is up-to-date on all patches and minimize the number of services running. Additionally,

[60] DameWare Development. DameWare NT Utilities 4.6. 22 Oct 2004. <http://www.majorgeeks.com/download2134.html>

[61] United States Computer Emergency Readiness Team. Vulnerability Note VU#909678: DameWare Mini Remote Control vulnerable to buffer overflow via specially crafted packets. 22 Dec. 2003 <http://www.kb.cert.org/vuls/id/909678>

[62] Connelly, Ken. [Intrusions] [LOGS] Summary of large-scale portscanning detects. 26 Apr 2004. <http://www.dshield.org/pipermail/intrusions/2004-April/007915.php>

[63] Nimda. 9 Jan 2004. <http://forum.webreseller.net/post-1471.html>

Deleted: → - → -

Deleted: - -

internal host 192.181.150.44 is responding on port 137. Port 137 has a long list of vulnerabilities. If absolutely needed port 137 should only be utilized within an intranet environment. Port 137 should be blocked both inbound and outbound .

Source Alert#11 → *192.181.11.7*

```
cat alertsfinal2 | grep "192.181.11.7" | egrep   -v
"169.254.0.0|169.254.25.129" | more

[**] SMB Name Wildcard [**] 192.181.11.7:137 -> 169.254.0.0:137
[**] SMB Name Wildcard [**] 192.181.11.7:137 -> 169.254.0.0:137
[**] SMB Name Wildcard [**] 192.181.11.7:137 -> 169.254.25.129:137
[**] SMB Name Wildcard [**] 192.181.11.7:137 -> 169.254.25.129:137
```

After the above filter grep is applied, there are only two alerts left out of 2000+. This device is trying to connect and cannot find a default gateway. I just went thru this with my brother's wireless router in his home network. The device is could be a Microsoft client, which cannot establish a default gateway because it is sending an incorrect wep key to the wireless AP. Depending on the universities available IT resources and/or student engagement policies regarding IT; the university personal should contact the owner to fix the problem, block the subnet (169.254.0.0)[64], or filter this alert. Issuing an `ipconfig /all` on the source host should look similar to the below.

```
      Connection-specific DNS Suffix  . :
      Description . . . . . . . . . . . : Microsoft Loopback Adapter

      Physical Address. . . . . . . . . : 02-00-4C-4F-4F-50
      Dhcp Enabled. . . . . . . . . . . : Yes
      Autoconfiguration Enabled . . . . : Yes
      Autoconfiguration IP Address. . . : 169.254.25.129
      Subnet Mask . . . . . . . . . . . : 255.255.0.0
      Default Gateway . . . . . . . . . :[65]
```

Internal source host 192.181.11.4 is also having this problem.

```
cat alertsfinal2 | grep "169.254.25.129" | egrep -v "192.181.11.7"
|more

[**] SMB Name Wildcard [**] 192.181.11.4:137 -> 169.254.25.129:137
[**] SMB Name Wildcard [**] 192.181.11.4:137 -> 169.254.25.129:137
```

Source Alert#12 → *69.138.77.62*

```
[**] 192.181.30.3 activity [**] 69.138.77.62:1033 -> 192.181.30.3:524
[**] 192.181.30.3 activity [**] 69.138.77.62:1064 -> 192.181.30.3:3019
```

Port 524 is NCP. This looks legitimate traffic and should be filtered.

[64] JANET. "Traffic which should be blocked by routers". <http://www.ja.net/CERT/JANET-CERT/prevention/cisco/private_addresses.html>

[65] "Solution Title: Win 98 PC as cable modem host to XP PC using router?". Expert-Exchange. 27 Dec 2003. <http://www.experts-exchange.com/Networking/Broadband/DSL_Cable/Q_20835464.html>

Deleted: → - → -

Deleted: - -

Source Alert#13 → *151.196.115.104*

Same as Source Alert#12.

Source Alert#14 → *64.12.24.35*

```
[**] High port 65535 TCP - possible Red Worm - traffic [**]
64.12.24.35:65535 -> 192.181.43.4:1214
[**] High port 65535 TCP - possible Red Worm - traffic [**]
192.181.43.4:1214 -> 64.12.24.35:65535

cat alertsfinal2 | grep  "64.12.24.35" | egrep  -v "192.181.43.8" |
egrep -v "192.181.43.4" |more
```

No results. This is not a virus. This is some file sharing going on with 2 hosts within the internal environment. Possibly one of the below:

TCP	kazaa	KAZAA file sharing app
udp	kazaa	KAZAA file sharing app
TCP	Morpheous	Morpheous file sharing app
udp	Morpheous	Morpheous file sharing app
TCP	Grokster	Grokster file sharing app
udp	Grokster	Grokster file sharing app

Source Alert#15 → *68.34.94.70*

Port 524 activities again. See Source Alert#14.

Source Alert#16 → *192.181.43.13*

```
[**] High port 65535 TCP - possible Red Worm - traffic [**]
205.188.5.100:65535 -> 192.181.43.13:1608

cat alertsfinal2 | grep "192.181.43.13" | egrep   -v
":1605|:1608|:1627|:1863" | more
```

Target ports thi s internal source is triggering are listed the egrep above. All target ports appear to be legitimate. This internal source looks like they are doing a lot of chatting. If this violates the university acceptable use policies, the user needs to be contact ed. If not, this alert needs to be filte red.

Port 1608

TCP	smart-lm	Smart Corp. License Manager

Deleted: → - → -

Deleted: - -

udp	smart-lm	Smart Corp. License Manager

Port 1605
Salutation Manager. See Source Alert#8.

Port 1627

TCP	t128-gateway	T.128 Gateway
udp	T128-gateway	T.128 Gateway

Port 1863

TCP	msnp	MSN Messenger Protocol
udp	msnp	MSN Messenger Protocol

Source Alert#17 → *68.33.49.146*

Most alerts are destine to port 51443. However, this external host appears to be firing additional activity alerts.

```
cat alertsfinal2  | grep "68.33.49.146" | egrep  -v ":51443" |more

04/22-20:06:02.593458  [**] 192.181.30.4 activity [**]
68.33.49.146:1041 -> 192.181.30.4:80
04/23-23:25:26.932914  [**] 192.181.30.4 activity [**]
68.33.49.146:2316
```

Port 2316 could be IM as noted in the Dshield reference below, but it is not a good practice to start basing analysis on ephemeral source ports. Port 1041 appears has no Dshield association. This is most likely a user browsing web pages. If 192.181.30.4 is truly a file server, I would question why it is running additional services? Hardware, in retrospect to the past 20 years, is relatively cheap. A file server should only run file server service, web server only web page service, a mail server only mail service, ect. As a defensive measure, services should be segmented to their own hardware. If a host is compromised, the impact to the university will be limited as the incident is mitigated through the incident handling process.

TCP	sent-lm	SENT License Manager
udp	sent-lm	SENT License Manager

Source Alert#18 → *67.167.3.240*

Port 2894. This is the external address in the Source Alert#9 conversation.

Source Alert#19 → *195.36.245.141*

Deleted: → - → -

Deleted: - -

```
[**] High port 65535 TCP - possible Red Worm - traffic [**]
192.181.153.81:1759 -> 195.36.245.141:65535
[**] High port 65535 TCP - possible Red Worm - traffic [**]
195.36.245.141:65535 -> 192.181.153.81:1759
```

IP Address: 195.36.245.141

HostName: f01m-6-141.d3.club-internet.fr

DShield Profile:

Country:	🇫🇷 FR
Contact E-mail:	abuse@club-internet.fr
AS Number:	5410
Total Records against IP:	not processed
Number of targets:	select update below
Date Range:	to

Update Summary

Whois:

```
% This is the RIPE Whois secondary server.
% The objects are in RPSL format.
%
% Rights restricted by copyright.
% See http://www.ripe.net/db/copyright.html

inetnum:      195.36.229.0 - 195.36.255.255
netname:      T-ONLINEFRANCE
descr:        Pools for ADSL customers
country:      FR
admin-c:      NOCT1-RIPE
tech-c:       NOCT1-RIPE
status:       ASSIGNED PA
notify:       ripe@t-online.fr
mnt-by:       T-ONLINEFRANCE
changed:      vox@t-online.fr 20021008
source:       RIPE

route:        195.36.128.0/17
descr:        T-Online France - Club Internet
origin:       AS5410
notify:       ripe@t-online.fr
mnt-by:       T-ONLINEFRANCE
changed:      vox@t-online.fr 20021009
source:       RIPE

role:         Network Operation Centre T-ONLINE FRANCE
address:      T-Online France - Club Internet
address:      11 rue de Cambrai
address:      75019 Paris
address:      France
phone:        +33 1 55 45 45 00
fax-no:       +33 1 55 45 47 78
e-mail:       ripe@t-online.fr
admin-c:      AV-RIPE
tech-c:       AV-RIPE
```

Deleted: → - → -

Deleted: - -

```
              tech-c:        OB346-RIPE
              tech-c:        DA3757-RIPE
              tech-c:        OT1274-RIPE
              nic-hdl:       NOCT1-RIPE
              mnt-by:        T-ONLINEFRANCE
              changed:       vox@t-online.fr 20040504
              source:        RIPE
```

This host does not show virus lik e symptoms. There is o nly one source and destination. Additionally, there is only one source port.

```
cat alertsfinal2 | grep "195.36.245.141" | egrep  -v ":1759" |more
```

No results.

TCP	spss-lm	SPSS License Manager
udp	spss-lm	SPSS License Manager

This appears to be a chat session going on between someone at the university and someone in France. The above alert could be filtered for target port 1759 , if this is acceptable use of the university network . An example filter would look like this:

```
(SRC host 1 95.36.245.141 and
 (
  (tcp and not dst port 1759 )
 )
```

Source Alert#20 → *24.43.50.166*

```
04/20-17:04:22.984513  [**] RFB - Possible WinVNC - 010708-1 [**]
24.43.50.166:3694 -> 192.181.82.2:5900
04/20-17:15:14.477356  [**] RFB - Possible WinVNC - 010708-1 [**]
192.181.82.2:5900 -> 24.43.50.166:3523
```

Source IP address (24.43.50.166) is from Canada.
Checking other WinVNC alerts:

```
cat alertsfinal2 | grep "WinVNC" | egrep  -v "24.43.50.166" | more
```

Not many results. WinVNC traffic primarily looks like it is be tween the above source and target. VNC should be blocked (port 5900) by the firewall. VNC traffic should not be allowed inside or outside of the university network. A business need for this terminal emulating needs to be established and an alternative m ethod of communication communicated to the users . If VNC is a preferred method, then an access list should be established on the either firewall or perimeter router for specific hosts. If remote management through VNC is not currently being performed by legitimate IT staff, this external address may have compromised several hosts within the university network.

Deleted: → - → -

Deleted: - -

IP Address: 24.43.50.166

HostName: CPE0010a4ebceb5 -CM.cpe.net.cable.rogers.com

DShield Profile:

Country:	🇨🇦 CA
Contact E-mail:	
AS Number:	812
Total Records against IP:	not processed
Number of targets:	select update below
Date Range:	to

Update Summary

Whois:

```
CustName:   Rogers Cable Inc. Lndn
Address:    1 Mount Pleasant Road
City:       Toronto
StateProv:  ON
PostalCode: M4Y-2Y5
Country:    CA
RegDate:    2003-08-20
Updated:    2004-09-28

NetRange:   24.43.48.0 - 24.43.51.255
CIDR:       24.43.48.0/22
NetName:    ON-ROG-LDN-18
NetHandle:  NET-24-43-48-0-1
Parent:     NET-24-43-0-0-1
NetType:    Reassigned
Comment:
RegDate:    2003-08-20
Updated:    2004-09-28

OrgAbuseHandle: RHI9-ARIN
OrgAbuseName:   Rogers High-Speed Internet
OrgAbusePhone:  +1-416-935-4729
OrgAbuseEmail:  abuse@rogers.com

OrgTechHandle: RHI9-ARIN
OrgTechName:   Rogers High-Speed Internet
OrgTechPhone:  +1-416-935-4729
OrgTechEmail:  abuse@rogers.com

# ARIN WHOIS database, last updated 2004-11-04 19:10
# Enter ? for additional hints on searching ARIN's WHOIS
database.


OrgName:    Rogers Cable Inc.
OrgID:      ROCA
Address:    One Mount Pleasant
City:       Toronto
StateProv:  ON
```

Deleted: → - → -

Deleted: - -

```
PostalCode: M4Y-2Y5
Country:    CA

NetRange:   24.43.0.0 - 24.43.255.255
CIDR:       24.43.0.0/16
NetName:    ROGERS-CAB-11
NetHandle:  NET-24-43-0-0-1
Parent:     NET-24-0-0-0-0
NetType:    Direct Allocation
NameServer: NS1.WLFDLE.RNC.NET.CABLE.ROGERS.COM
NameServer: NS1.YM.RNC.NET.CABLE.ROGERS.COM
Comment:
RegDate:
Updated:    2004-10-18

OrgAbuseHandle: RHI9-ARIN
OrgAbuseName:   Rogers High-Speed Internet
OrgAbusePhone:  +1-416-935-4729
OrgAbuseEmail:  abuse@rogers.com

OrgTechHandle: RHI9-ARIN
OrgTechName:   Rogers High-Speed Internet
OrgTechPhone:  +1-416-935-4729
OrgTechEmail:  abuse@rogers.com

# ARIN WHOIS database, last updated 2004-11-04 19:10
# Enter ? for additional hints on searching ARIN's WHOIS
database.


OrgName:    Rogers Cable Inc.
OrgID:      ROCA
Address:    One Mount Pleasant
City:       Toronto
StateProv:  ON
PostalCode: M4Y-2Y5
Country:    CA
Comment:
RegDate:
Updated:    2003-07-08

AbuseHandle: RHI9-ARIN
AbuseName:   Rogers High-Speed Internet
AbusePhone:  +1-416-935-4729
AbuseEmail:  abuse@rogers.com

AdminHandle: IPMAN-ARIN
AdminName:   IP Management
AdminPhone:  +1-416-935-4729
AdminEmail:  ipmanage@rogers.wave.ca

TechHandle: RHI9-ARIN
TechName:   Rogers High-Speed Internet
TechPhone:  +1-416-935-4729
TechEmail:  abuse@rogers.com
```

Deleted: → - → -

Deleted: - -

```
# ARIN WHOIS database, last updated 2004-11-04 19:10
# Enter ? for additional hints on searching ARIN's WHOIS
database.
```

Top 20 Destination IPs from SnortSnarf Output.

This page provides summary information about alerts acquired using input module SnortFileInput, with sources:

- alertsfinal2

The most active destination IPs are shown. Rank is determined by the number of alerts with that IP as the destination. Within a rank, IPs are sorted by # of signatures, then by IP number.

Rank	Total # Alerts	**Destination IP**	# Signatures triggered	Originating sources
Dest #1	35300 alerts	**192.181.30.4**	12 signatures	(313 source IPs)
Dest #2	15905 alerts	**192.181.30.3**	5 signatures	(199 source IPs)
Dest #3	3435 alerts	**192.181.43.8**	4 signatures	(9 source IPs)
Dest #4	3067 alerts	**64.12.24.34**	1 signatures	(3 source IPs)
Dest #5	2989 alerts	**67.167.3.240**	1 signatures	192.181.69.232
Dest #6	2603 alerts	**210.120.128.117**	1 signatures	192.181.11.4
Dest #7	2165 alerts	**64.12.24.35**	1 signatures	192.181.43.4, 192.181.43.8
Dest #8	2160 alerts	**192.181.97.43**	4 signatures	130.79.183.1, 209.164.32.205
Dest #9	2120 alerts	**192.181.43.13**	2 signatures	(7 source IPs)
Dest #10	1808 alerts	**192.181.97.55**	8 signatures	209.164.32.205, 216.109.117.108
Dest #11	1529 alerts	**169.254.0.0**	3 signatures	(4 source IPs)
Dest #12	1526 alerts	**192.181.69.232**	7 signatures	(10 source IPs)
Dest #13	1390 alerts	**192.181.153.81**	2 signatures	(4 source IPs)
Dest #14	1280 alerts	**192.181.17.4**	2 signatures	(23 source IPs)
Dest #15	1215 alerts	**169.254.25.129**	1 signatures	192.181.11.7,

Deleted: → - → -

Deleted: - -

				192.181.11.4
Dest #16	1195 alerts	24.43.50.166	3 signatures	(11 source IPs)
Dest #17	1160 alerts	192.181.17.3	1 signatures	(24 source IPs)
Dest #18	1071 alerts	192.181.53.10	1 signatures	(20 source IPs)
Dest #19	1043 alerts	192.181.53.84	5 signatures	(22 source IPs)
Dest #20	883 alerts	195.36.245.141	1 signatures	192.181.153.81

Destination Alert#1 → *192.181.30.4*

A large part of both Destination Alert#1 and Alert#2 is port 51443 and 8009. These ports have been addressed in several Practicals .[66] The below grep was entered to filter conversations based on port 51443 , 80 and port 8009 in order to see what other alerts this host might be firing.

```
cat alertsfinal2 | grep "192.181.30.4" | egrep  -v
":51443|:8009|:80" | more
```

Source Alert#3 or 209.164.32.205 is prevalent and explained above .

Destination Alert#2 → *192.181.30.3*

Same 51443 traffic the majority of alerts to this in ternal address . What is left after filtering?

```
cat alertsfinal2 | grep "192.181.30.3" | egrep  -v
":51443|:524|:80" | more
```

```
[**] 192.181.30.3 activity [**] 156.17.186.27:1631 -> 192.181.30.3:2745
[**] 192.181.30.3 activity [**] 156.17.186.27:1631 -> 192.181.30.3:2745
[**] 192.181.30.3 activity [**] 156.17.186.27:1636 -> 192.181.30.3:6129
[**] 192.181.30.3 activity [**] 156.17.186.27:1631 -> 192.181.30.3:2745
[**] 192.181.30.3 activity [**] 156.17.186.27:1636 -> 192.181.30.3:6129
```

IP Address: 156.17.186.27

HostName: arka27.ar.wroc.pl

DShield Profile:

Country:	🟥 PL
Contact E-mail:	skowr@WASK.WROC.PL
AS Number:	8970

[66] Kroeger, Tim. "Security Information Management Systems: What are they? Who makes them? Do I need one?". 18 May 2004. <http://www.giac.org/practical/GCIA/Tim_Kroeger_GCIA.pdf>

Deleted: → - → -

Deleted: - -

This host is sending tragic destine to known Bagle.C po rt or 2745 and Dameware port 6129. Next was to check how many more alerts on port 6129 and if any host have responded to port 6129 traffic.

```
cat alertsfinal2 | grep ":6129" | more
```

There are quite a few, but no internal host has responded or has sent any traffic destine to target port 6129. This is good. The university appears to be Bagle.C free.

```
cat alertsfinal2 | grep ":6129" | wc  -l
165
```

Still several Bagle.C externally infected hosts. But the university appears to have the defensive measures in pla ce which prevent internal hosts from becoming infected.

Additionally, there is traffic destine to port 3019 from a two external source addresses. This is a prevalent event with 1393 alerts. Both 151.196.115.104 and 69.138.77.62 are also showing activity alerts on port 524. The nature of this traffic appears legitimate based on frequency and no additional alerts. These alerts need to be filtered.

```
cat alertsfinal2 | grep "192.181.30.3:3019" | wc  -l
1393
```

TCP	resource_mgr	Resource Manager
udp	resource_mgr	Resource Manager

Destination Alert#3 and Source Alert#6 → *192.181.43.8*

No other hosts have a source port of 1971. External address 66.136.201.252 scanned internal host 192.181.43.8 . Additionally, output shows the 7 destination IP addresses being ta rgeted by this host. This internal host is extensively using ephemeral source ports 1969, 1970, 1971, 1972 and 1979.

```
cat alertsfinal2 | grep "192.181.43.8" | egrep  -v
":1971|:1979|:65535|:1972|:1973" | more

[**] [UMBC NIDS IRC Alert] IRC user /kill detected, possible trojan.
[**] 64.124.166.200:6667 -> 192.181.43.8:1381
[**] Null scan! [**] 66.136.201.252:0 -> 192.181.43.8:0
[**] Null scan! [**] 66.136.201.252:0 -> 192.181.43.8:0
[**] Null scan! [**] 66.136.201.252:0 -> 192.181.43.8:0
[**] [UMBC NIDS IRC Alert] IRC user /kill detected, possible trojan.
[**] 64.124.166.200:6667 -> 192.181.43.8:1391
```

All External IP a ddresses appear to be legitimate United States assigned IPs. 64.124.166.200 machine sent IRC Kill to the internal destination. There are

Deleted: → - → -

Deleted: - -

various Trojans this machine could be infected with. This could be OpC BO, which uses the ports above.[67] This system is definitely not healthy. A quick search of Google for the above ports returned the below possible exploits , including two references to Ba ckDoor.Bifrose.

Zspy
http://www.pestpatrol.com/zks/pestinfo/z/zspy_ii_0_99b.asp

BackDoor.Bifrose
http://securityresponse.symantec.com/avcenter/venc/data/pf/backdoor.bifrose.html
http://forum.gladiator -antivirus.com/index.php?showtopic=19278

Above shows a target port of 1381 which is registered as Apple Network License Manager. Possibly an apple computer or maybe could be running iPod software. Regardless, this internal host should be investigated. Check for updated virus software and run a scan.

Destination Alert#4 → *64.12.24.34*

```
[**] High port 65535 TCP - possible Red Worm - traffic [**]
64.12.24.34:65535 -> 192.181.43.13:1605
```

Port 1605

TCP	slp	Salutation Manager (Salutation Protocol)
udp	slp	Salutation Manager (Salutation Protocol)

All three addresses listed in the grep below appear to have Salutation M anager installed.[68]

```
cat alertsfinal2  | grep "64.12.24.34" | egrep  -v
"192.181.43.4|192.181.43.13|192.181.43.8" | more
```
No Results.

After reading more about Salutation Manager, this traffic seems legit imate. If this software does not violate any application usage policies and is confirmed to be the application generating the alerts, the alerts should be filtered.[69]

There is also 2063 High port 65536 Red Worm alerts associated with target port 2718. After closer analysis, this does not appear to be virus related.

```
[**] High port 65535  TCP - possible Red Worm  - traffic [**]
64.12.24.34:65535  -> 192.181.43.8:2718
```

[67] G-Lock Software. "OpC BO". Apr 1999. <http://www.glocksoft.com/trojan_list/OpC_BO.htm>
[68] The Salutation Consortium. "Salutation personalities". 2004. <http://www.salutation.org/techtalk/person.htm>
[69] Spectrum Online. "Building Networks on the Fly". 2004. <http://www.spectrum.ieee.org/WEBONLY/publicfeature/mar01/net.html>

Deleted: → - → -

Deleted: - -

```
[**] High port 65535  TCP - possible Red Worm  - traffic [**]
192.181.43.8:2718  -> 64.12.24.34: 65535

cat alertsfinal2 | grep "64.12.24.34" | grep ":2718" | wc   -l
   2063
```

Port 2718 is associated with PN Requester 2, which has no known vulnerabilities. I am unsure why 64.12.24.34 insists on transmitting all of its request to on a high ephemeral por t of 65535. This host may be mis-configured. If the mis-configuration cannot be resolved, I would recommend filtering this alert for host 64.12.24.34.

Destination Alert#5 and Source Alert#9 → *67.167.3.240*

See Source Alert#9 for analysis.

Destination Alert#6 and Source Alert#7 → *210.120.128.117*

```
cat alertsfinal2 | grep "210. 120.128.117" | egrep  -v ":137"| more
```

and

```
cat alertsfinal2 | grep "210.120.128.117" | egrep   -v
"192.181.11.4" |more
```

Both return no results. All alerts from the above external address ar e:

```
04/22-13:31:09.983720 [**] SMB Name Wildcard [**] 192.181.11.4:137 -> 210.120.128.117:137
04/22-13:31:11.481648 [**] SMB Name Wildcard [**] 192.181.11.4:137 -> 210.120.128.117:137
04/22-13:31:12.981680 [**] SMB Name Wildcard [**] 192.181.11.4:137 -> 210.120.128.117:137
04/22-13:31:37.585064 [**] SMB Name Wildcard [**] 192.181.11.4:137 -> 210.120.128.117:137
```

Destination host is from Korea. This looks like a compromised host. Why is the university allowing port 137 (typically associated with NetBio s) outbound? This should be blocked at the firewall to resolve this alert.

IP Address: 210.120.128.117
HostName: 210.120.128.117
DShield Profile:

Country:	🇰🇷 KR
Contact E-mail:	abuse@bora.net
AS Number:	3786

Destination Alert#7 and Source Alert#14 → *64.12.24.35*

```
[**] High port 65535 TCP - possible Red Worm - traffic [**]
64.12.24.35:65535 -> 192.181.43.4:1214
[**] High port 65535 TCP - possible Red Worm - traffic [**]
192.181.43.4:1214 -> 64.12.24.35:65535
```

TCP	kazaa	KAZAA file sharing app
udp	kazaa	KAZAA file sharing app

Deleted: → - → -

Deleted: - -

TCP	Morpheous	Morpheous file sharing app
udp	Morpheous	Morpheous file sharing app
TCP	Grokster	Grokster file sharing app
udp	Grokster	Grokster file sharing app

This host is also running academic related software (NetOp School), but triggering high port virus alert. [70] This should be filtered after confirmation of legitimate traffic.

```
[**] High port 65535 TCP - possible Red Worm - traffic [**]
64.12.24.35:65535 -> 192.181.43.8:1971
[**] High port 65535 TCP - possible Red Worm - traffic [**]
192.181.43.8:1971 -> 64.12.24.35:65535
```

Destination Alert#8 → *192.181.97.43*

```
[**] Null scan! [**] 209.164.32.205:0 -> 192.181.97.43:0
[**] Tiny Fragments - Possible Hostile Activity [**] 209.164.32.205 ->
192.181.97.43

cat alertsfinal2 | grep "192.181.97.43" | egrep -v "209.164.32.205" |
more
```

Only 1 Exploit x86 NOOP alert. This internal host appears t o be a victim of Source Alert#3 and Destination Alert#10 f or analysis.

Destination Alert#9 and Source Alert#16 → *192.181.43.13*

See Source Alert#16.

Destination Alert#10 → *192.181.97.55*

```
cat alertsfinal2 | grep "192.181.97.55" | more

[**] Null scan! [**] 209.164.32.205:0 -> 192.181.97.55:0
[**] Tiny Fragments - Possible Hostile Activity [**] 209.164.32.205 ->
192.181.97.55

cat alertsfinal2 | grep  -c "192.181.97.55"
1811
```

[70] NetOp School. "Software for Networked Classrooms". 2004. <http://www.crossteccorp.com/netopschool/>

Deleted: → - → -

Deleted: - -

A lot of alerts from external source host 209.164.32.205. This could be a victim of Source Alert#3. Taking a closer look at the external host , they appear to be nmap scanning several hosts within the university environment.

```
cat alertsfinal2 | grep  -c "209.164.32.205"
4770
```

Besides scanning has the Target hosts triggered any other alerts ? It appears 192.181.97.55 was only subject to being scan ned by 209.164.32.205, but internal host 192.181.97.43 may not have been so lucky.

```
cat alertsfinal2 | grep "209.164.32.205" | egrep   -v "Tiny|Null"
|more

[**] Probable NMAP fingerprint attempt [**] 209.164.32.205:0 ->
192.181.81.116:0
[**] SYN-FIN scan! [**] 209.164.32.205:22924 -> 192.181.97.43:6625
```

A SYN-FYN was sent to Destination Alert#8 by 209.164.32.205. Even though 192.181.97.43 has not triggered any response alerts, this server should be investigated depending on its criticality within the university network. All SYN - FIN should be carefully scrutinized. Including the below:

```
cat alertsfinal2 | grep "SYN -FIN" | more
[**] SYN-FIN scan! [**] 209.164.32.205:22924 -> 192.181.97.43:6625
[**] SYN-FIN scan! [**] 61.48.8.56:53558 -> 192.181.112.209:1773
[**] SYN-FIN scan! [**] 61.48.8.56:60699 -> 192.181.112.209:16026
```

192.181.112.209 may be compromised based on the additional alerts below based on the chronological time of the alerts. It appears 61.48.8.56 scanned internal source host 192.181.112.209, t hen sent a SYN -FIN, then the does a TFTP connection. It is worth investigating the internal host.

```
[**] Null scan! [**] 61.48.8.56:0 -> 192.181.112.209:0
[**] TFTP - External TCP connection to internal tftp server [**]
61.48.8.56:65242 -> 192.181.112.209:69
```

Dshield shows IP 61.48.8.56 resolves to the below, definitely a suspicious external host. To defend against this attack, TFTP port 69 should be blocked inbound. TFTP requires no name or user password. Running quick check, I found it doesn't appear t o be blocked based on the below grep results. Also 192.181.190.91 may be compromised .

```
cat alertsfinal2 | grep ":69" | more
[**] TFTP - Internal UDP connection to external tftp server [**]
192.181.190.91:1036 -> 68.160.1.135:69
```

Country:	🇨🇳 CN
Contact E-mail:	abuse@cnc-noc.net

Deleted: → - → -

Deleted: - -

AS Number:	4814
Total Records against IP:	14
Number of targets:	6
Date Range:	2004-08-23 to 2004-10-22

```
% [whois.apnic.net node-1]
% Whois data copyright terms
http://www.apnic.net/db/dbcopyright.html
inetnum:       61.48.0.0 - 61.51.255.255
netname:       CNCGROUP-BJ
descr:         CNCGROUP Beijing province network
descr:         China Network Communications Group Corporation
descr:         No.156,Fu-Xing-Men-Nei Street,
descr:         Beijing 100031
country:       CN
admin-c:       CH455-AP
tech-c:        SY21-AP
mnt-by:        APNIC-HM
mnt-lower:     MAINT-CNCGROUP-BJ
changed:       hm-changed@apnic.net 20031017
status:        ALLOCATED PORTABLE
source:        APNIC
role:          CNCGroup Hostmaster
e-mail:        abuse@cnc-noc.net
address:       No.156,Fu-Xing-Men-Nei Street,
address:       Beijing,100031,P.R.China
nic-hdl:       CH455-AP
phone:         +86-10-68019956
fax-no:        +86-10-68019958
country:       CN
admin-c:       CH444-AP
tech-c:        CH444-AP
changed:       abuse@cnc-noc.net 20031016
mnt-by:        MAINT-CNCGROUP
source:        APNIC
person:        sun ying
address:       Beijing Telecommunication Administration
address:       TaiPingHu DongLi 18, Xicheng District
address:       Beijing 100031
country:       CN
phone:         +86-10-66198941
fax-no:        +86-10-68511003
e-mail:        suny@publicf.bta.net.cn
nic-hdl:       SY21-AP
mnt-by:        MAINT-CHINANET-BJ
changed:       suny@publicf.bta.net.cn 19980824
source:        APNIC
```

Destination Alert#11 → *169.254.0.0*

See Source Alert# 11

Destination Alert#12 → *192.181.69.232*

Deleted: → - → -

Deleted: - -

See Source Alert#9

Destination Alert#13 → *192.181.153.81*

See Source Alert#19

Destination Alert#14 → *192.181.17.4*

Possible compromised internal or extensive IRC usage from this internal host. Sending the user a e -mail message with all 1310 Snort alerts may get the user to clean up his system, or look to see if the system has been compromised.

```
cat alertsfinal2 | grep "192.181.17.4" | egrep  -v "EXPLOIT x86
NOOP" | more

[**] [UMBC NIDS IRC Alert] Possible drone command detected. [**]
130.74.159.212:7000 -> 192.181.17.45:2162
[**] [UMBC NIDS IRC Alert] Possible sdbot floodnet detected attempting
to IRC [**] 192.181.17.45:1029 -> 164.15.194.17:7000
[**] [UMBC NIDS IRC Alert] IRC user /kill detected, possible trojan.
[**] 131.96.20.15:7000 -> 192.181.17.45:1054
```

TCP	afs3-fileserver	file server itself msdos
TCP	ExploitTranslation	[trojan] Exploit Translation Server
TCP	ExploitTranslation	[trojan] Exploit Translation Server
TCP	Kazimas	[trojan] Kazimas
TCP	RemoteGrab	[trojan] Remote Grab
TCP	SubSeven2.1Gold	[trojan] SubSeven 2.1 Gold
TCP	SubSeven	[trojan] SubSeven
udp	afs3-fileserver	file server itself

Destination Alert#15 → *169.254.25.129*

See Source Alert#11

Destination Alert#16 → *24.43.50.166*

See Source Alert#20

Deleted: → - → -

Deleted: - -

Destination Alert#17 → *192.181.17.3*

```
cat alertsfinal2 | grep "192.181.17.3" | egrep -v ":80" |more
```

No results. This host is only triggering the below alert from a legitimate source host. There is no return traffic alert from 192.181.17.3. This could be a web server creating false positive alerts. Further inve stigations is needed to confirm this alert to be a false positive. Since the below external host below belongs to IBM this is probably legitimate traffic (unless spoofed). T his alert should be filtered for the below source address.

```
[**] EXPLOIT x86 NOOP [**] 129.33.49.251:34537 -> 192.181.17.3:80
```

Country:	🇺🇸 US
Contact E-mail:	noc@btv.ibm.com

Destination Alert#18 → *192.181.53.10*

```
cat alertsfinal2 | grep "192.181.53.10  " | egrep -v ":80" | more
```

No results. It appears this maybe the external webserver since no other traffic beside port 80 are logged. This host is only firing the below alerts to the various external hosts. This is most likely a false -positve, but cannot be confirmed without further analysis.

```
[**] EXPLOIT x86 NOOP [**] 211.99.126.60:3170 -> 192.181.53.10:80
```

Destination Alert#19 → *192.181.53.84*

```
cat alertsfinal2 | grep "192.181.53.84" | egrep   -v ":80" | more
```

No results. This host is only firing the bel ow alert from various sources. Most external source hosts are geographically located in the US and are only triggering the EXPLOIT x86 NOOP alert . This is a likely false positive, but a more thorough analysis will need to be completed before this prelimi nary guess can be solidified.

```
[**] EXPLOIT x86 NOOP [**] 68.228.103.33:2278 -> 192.181.53.84:80
```

Destination Alert#20 → *195.36.245.141*

See Source Alert#19

External Addresses Under Consideration

Registration information for these addresses and reasons why they were chosen are listed under the appropriate alert heading .

Source Alert#3 → *209.164.32.205*

Source Alert#10 → *220.197.192.39*

Deleted: → - → -

Deleted: - -

Source Alert#19 → *195.36.245.141*
Source Alert#20 → *24.43.50.166*
Within Destination Alert#10 → *61.48.8.56*

Additional External Threats:

Spam from Denmark

```
cat alertsfinal2 | grep ":1024" | more
[**] EXPLOIT x86 NOOP [**] 80.138.145.244:1024 -> 192.181.24.31:80
```

Nterm pop-up spam from Netherlands

```
cat alertsfinal2 | grep ":1026" | more
[**] EXPLOIT x86 NOOP [**] 82.217.220.171:1026 -> 192.181.70.53:80
```

External host located in Mexico infected with the Kunag2TheVirus. Targeting port 17300, two hosts. Nothing you can do about this activity except refine the snort rule to trigger on something besides the activity alert.

```
cat alertsfinal2 | grep ":17300"
[**] 192.181.30.4 activity [**] 201.128.103.27:3539 -
>192.181.30.4:17300
[**] 192.181.30.3 activity [**] 201.128.103.27:3538 -
>192.181.30.3:17300
```

192.181.1.3 is the external DNS server for the university. There are a lot of alerts triggering on legitimate DNS traffic from geographically nearby external source hosts. The below alerts should be filtered.

```
[**] NMAP TCP ping! [**] 63.211.17.228:80 -> 192.181.1.3:53
```

Since this DNS server is critical to our infrastructure. The below alerts should be investigated. 65.248.229.131 is registered to Swartz Paper company. [71] The university might have something printed by the company, but I do not know why they would need to be doing a TFTP to the DNS server. Need a Hex dump with ASCI translation of the below alerts for further analysis.

```
[**] TFTP - Internal UDP connection to external tftp server [**]
208.178.209.4:69 -> 192.181.1.3:53

[**] TFTP - Internal UDP connection to external tftp server [**]
65.248.229.131:69 -> 192.181.1.3:53
```

Potentially Compromised Internal Hosts

[71] Industry Center - Containers & Packaging. "Schwarz Paper Company Profile". 2004. <http://biz.yahoo.com/ic/44/44506.html>

Deleted: → - → -

Deleted: - -

192.181.150.44 is communicating with multiple foreign countries across port 137 and 80. The host is also triggering SMB Name Wildcard and EXPLOIT x86 NOOP. One of the destination IP addresses on port 137 is 208.182.190.91, which is also sending port 4000 requests to 192.181.40.3, 4. This is a common Trojan port.

The below host are all trying to communicate with external sources through TFTP, with the exception of 192.181.190.91 which was targeted using TFTP.

192.181.190.91 - subject to External TFTP after scan from China host .
192.181.111.34 - TFTP to Denmark external host
192.181.69.232 – TFTP to US external host
192.181.80.44 – TFTP to US AOL
192.181.27.232 – TFTP to Australia
192.181.75.84 – TFTP to Netherlands

Destination Alert#14 → 192.181.17.4 – possible IRC Trojan.

This host has been heavily scanned and targeted by external Chinese source.
192.181.112.209 SYN -FIN Scan

Nimda infected hosts
192.181.97.85
192.181.97.146

Subseven – The below hosts are targeting port 27374 1 and triggering the below alert:

```
[**] Possible trojan server activity [**] 65.34.25.206:27374 ->
192.181.24.44:80
```

192.181.24.44
192.181.6.7
192.181.24.34
192.181.12.6
192.181.5.44
192.181.34.11
192.181.18.206
192.181.60.14
192.181.15.255
192.181.12.7
192.181.24.74

The source is in the Netherlands targeting a single internal host. The host should be checked for Back Orifice.

```
Back Orifice [**] 217.69.156.193:34002 -> 192.181.153.143:31337
```

Deleted: → - → -

Deleted: - -

References

1. <http://isc.sans.org/logs/raw/ file 2002.6.11>
2. 2 Oct 2004. <http://netflow3.nhltc.edu.tw/netflow/scan/c6509/2004/2004-10/2004-10-02/2004-10-02.202131-OUT.html>
3. Abacus. "Performance Assurance". 2004. <http://www.abacus-engr.com/services/performance_assurance/remote_diagnostics.html>
4. Abacus. "Higher Education Facilities". <http://www.abacus-engr.com/portfolio/education2/index-2.html>
5. American Registry for Internet Numbers. <http://ws.arin.net/cgi-bin/whois.pl>
6. ArcSight. "Security Information Management Software for the Enterprise". <http://www.arcsight.com>
7. Bakos, George. TCPDUMP Public Repository: Re: [TCPdump-workers] understanding filtering. 17 Dec 2002. <http://www.TCPdump.org/lists/workers/2002/12/msg00088.html>
8. Bauer, Mick. Linux Journal: Issue 85: Paranoid Penguin: Checking Your Work with Scanners, Part I (of II): nmap. 1 May 2001. <http://www.linuxjournal.com/article.php?sid=4561>
9. Bauer, Mick. Linux Journal: Paranoid Penguin: Checking Your Work with Scanners, Part I (of II): nmap. 1 May 2001. <http://www.linuxjournal.com/article.php?sid=4561>
10. Breault, Steve. "SANS Intrusion Detection in Depth". 2004. <http://www.giac.org/practical/GCIA/Steve_Breault_GCIA.pdf>
11. Christiansen, Chris and Kolodgy, Charles. "ArcSight Vendor Profile: Seeing Through the Clutter". Feb 2002. <http://www.arcsight.com/graphics/news/updated%20IDC%20report.pdf>
12. Cisco Systems. <http://www.cisco.com/en/US/customer/products/hw/switches/ps708/products_configuration_guide_chapter09186a008007f4c4.html>
13. Cisco Systems. <http://www.cisco.com/en/US/customer/products/hw/vpndevc/ps4077/products_qanda_item09186a008017f8e4.shtml>
14. Cisco Systems. <http://www.cisco.com/en/US/customer/products/hw/cable/ps2217/products_feature_guide_chapter09186a008019b571.html>
15. Cisco Systems. "CISCO IDS 4200 SERIES SENSORS". 2004. <http://www.cisco.com/en/US/products/hw/vpndevc/ps4077/index.html>
16. Cisco Systems. "CISCO IDS 4200 SERIES SENSORS". 2004. <http://www.cisco.com/en/US/products/hw/vpndevc/ps4077/products_qanda_item09186a008017f8e4.shtml>
17. Cisco Systems. "Cisco Intrusion Detection System Appliance and Module Installation and Configuration Guide Version 4". 2003. <http://www.cisco.com/application/pdf/en/us/guest/products/ps5398/c1676/ccmigration_09186a00801a24ce.pdf>
18. Cisco Systems. "Cisco Intrusion Detection System Appliance and Module Installation and Configuration Guide Version 4". 2003. <http://www.cisco.com/en/US/products/sw/secursw/ps2113/products_installation_and_configuration_guide_chapter09186a008014a238.html>
19. Cisco Systems. "Cisco Secure Intrusion Detection System Version 2.2.0 Release Notes". 1998. <http://www.cisco.com/en/US/products/sw/secursw/ps5052/prod_release_note09186a00800ee999.html>
20. Cisco Systems. "Maintaining Security Monitor". <http://www.cisco.com/en/US/products/sw/cscowork/ps3991/products_user_guide_chapter09186a008018d96f.html>
21. Cisco Systems. "Release Notes for Cisco Secure Policy Manager Version 2.3.3i". Jan 2002. <http://www.cisco.com/en/US/products/sw/secursw/ps2133/prod_release_note09186a00800d9cc2.html>

Deleted: → - → -

Deleted: - -

22. Common Vulnerabilities and Exposures. CAN-1999-0523. 26 Jul 1999. <http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0523>
23. Common Vulnerabilities and Exposures. CAN-1999 -0635. 8 Aug 1999. <http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0635>
24. Common Vulnerabilities and Exposures. CAN-1999-1373. 12 Sept 2001. <http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-1373>
25. Common Vulnerabilities and Exposures. CVE-2000-0324. 9 Mar 2002. <http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2000-0324>
26. Computer Associates. eTrust PestPatrol Pest Encyclopedia: Backweb. 26 Oct 2004. <http://www.pestpatrol.com/pestinfo/b/bsackweb.ap>
27. Connelly, Ken. [Intrusions] [LOGS] Summary of large-scale portscanning detects. 26 Apr 2004. <http://www.dshield.org/pipermail/intrusions/2004-April/007915.php>
28. DameWare Development. DameWare NT Utilities 4.6. 22 Oct 2004. <http://www.majorgeeks.com/download2134.html>
29. Dickey, Neil Ph.D. Security Focus Incident Archive: <http://www.securityfocus.com/archive/75/319981/2003-04-24/2003-04-30/0>
30. DShield. Port Report: 2161. 27 Oct 2004. <http://www.dshield.org/port_report.php?port=2161&recax=1&tarax=2&srcax=2&percent=N&days=40&Redraw=Submit+Query>
31. DShield. Port Report: 80. 27 Oct 2004. <http://www.dshield.org/port_report.php>
32. Expert-Exchange. "Solution Title: Win 98 PC as cable modem host to XP PC using router?". 27 Dec 2003. <http://www.experts-exchange.com/Networking/Broadband/DSL_Cable/Q_20835464.html>
33. Fischer, Amy. "Network taps enable passive monitoring". 28 Oct 2002. <http://www.nwfusion.com/news/tech/2002/1028techupdate.html>
34. fyodor@insecure.org. "Remote OS detection via TCP/IP Stack FingerPrinting". 11 Jun 2002. <http://www.insecure.org/nmap/nmap-fingerprinting-article.html>
35. G-Lock Software. "OpC BO". Apr 1999. <http://www.glocksoft.com/trojan_list/OpC_BO.htm>
36. IEEE. 2004. <http://standards.ieee.org/regauth/oui/oui.txt>
37. Industry Center - Containers & Packaging. "Schwarz Paper Company Profile". 2004. <http://biz.yahoo.com/ic/44/44506.html>
38. JANET. "Traffic which should be blocked by routers". <http://www.ja.net/CERT/JANET-CERT/prevention/cisco/private_addresses.html>
39. Janowski, Mike, Oele, Tom, and Shipley, Greg. "Too Much Information " 12 Sep 2003. <http://nwc.securitypipeline.com/showArticle.jhtml;jsessionid=K1PQCNLF2TWFCQSNDBGCKHQ?articleId=14700464&printableArticle=true>
40. Janowski, Mike, Oele, Tom, and Shipley, Greg. "Too Much Information " 12 Sep 2003. <http://nwc.securitypipeline.com/showArticle.jhtml;jsessionid=K1PQCNLF2TWFCQSNDBGCKHQ?articleId=14700464&printableArticle=true>
41. Kozierok, Charles M. "IP Datagram General Format" Version 2.0. 7 Jun 2004. <http://www.TCPipguide.com/free/t_IPDatagramGeneralFormat.htm>
42. Kozierok, Charles M. 'TCP Checksum Calculation and the TCP "Pseudo Header"' Version 2.0. 7 Jun 2004. <http://www.TCPipguide.com/free/t_TCPChecksumCalculationandtheTCPPseudoHeader-2.htm>
43. Kroeger, Tim. "Security Information Management Systems: What are they? Who makes them? Do I need one?". 18 May 2004. <http://www.giac.org/practical/GCIA/Tim_Kroeger_GCIA.pdf>
44. Kroeger, Tim "GCIA Practical". 2003. <http://www.giac.org/practical/GCIA/Tim_Kroeger_GCIA.pdf>
45. Leu, Matthias. "News January 1999". 12 Feb 1999. <http://www.leu.de/security/0199_e.html>
46. Lindsey, Mark R. University of North Carolina, Department of Computer Science: Work Log: Wed Jan 29 10:57:08 EST 2003. 29 Jan 2003. <http://www.cs.unc.edu/~lindsey/7ds/log/>
47. Lucent Technologies. "Award-winning, performance-proven cost-saving software" <http://www.lucent.com/solutions/netops_enter.html>

Field Code Changed

Deleted: → - → -

Deleted: - -

48. Mason, Dave. Segment 7. 5 Jun 2004. <http://www.davemason.com/june04.html>
49. Nathan, Jeff and Caswell, Brian. 100Mb IDS Tapping Diagram (with only 100bt span port). <http://www.snort.org/docs/100Mb_tapping1.pdf>
50. NetOp School. "Software for Networked Classrooms". 2004. <http://www.crossteccorp.com/netopschool/>
51. Niem, Tu. "Intrusion Detection in Depth". Pgs 68-70. 23 Jan 2003. <http://www.giac.org/practical/GCIA/Tu_Niem_GCIA.pdf>
52. Nimda. 9 Jan 2004. <http://forum.webreseller.net/post-1471.html>
53. nsck2000@yahoo.com. DSHIELD: LOGS: GIAC GCIA Version 3.3 Practical Detect(s) b. 31 Aug 2002. <http://www.dshield.org/pipermail/intrusions/2002-August/005072.php>
54. *procana@insight.rr.com.* Neohapsis: Re: [Snort-users] (snort_decoder): Truncated TCP Options. 27 Apr 2003. <http://archives.neohapsis.com/archives/snort/2003-04/1176.html>
55. *procana@insight.rr.com.* Neohapsis: Re: [Snort-users] (snort_decoder): Truncated TCP Options. 27 Apr 2003. <http://archives.neohapsis.com/archives/snort/2003-04/1176.html>
56. Red Hat. Errata: Updated ucd-snmp packages available. 12 Mar 2002. <http://rhn.redhat.com/errata/RHSA-2001-163.html>
57. Roesch, Martin and Poor, Mike. Track 3: Intrusion Detection In-Depth, Network Traffic Analysis using TCPdump. Parts 1 and 2 (slide 3-34). 2004.
58. The Salutation Consortium. "Salutation Personalities". 2004. <http://www.salutation.org/techtalk/person.htm>
59. Shipley, Greg and Miller, Patrick. "Cisco's NIDS Solution Grows Up". 21 Oct 2002. <http://www.networkcomputing.com/1322/1322sp3.html>
60. Snort Signature Database. "Building Networks on the Fly". 2004. <http://www.snort.org/snort-db/sid.html?sid=116-55>
61. Spectrum Online. "Building Networks on the Fly". 2004. <http://www.spectrum.ieee.org/WEBONLY/publicfeature/mar01/net.html>
62. Spyware-Removal. "BackWeb Removal How to remove and uninstall backweb adware". 2004. <http://www.spysweeper.com/backweb-removal.html>
63. Storm, Peter H. "GIAC Certified Intrusion Analyst (GCIA) Practical Assignment Version 3.3". 15 Nov 2003. <http://www.giac.org/practical/GCIA/Pete_Storm_GCIA.pdf>
64. The Swiss Education & Research Network. "Default TTL Values in TCP/IP". 2004. <http://secfr.nerim.net/docs/fingerprint/en/ttl_default.html>
65. Trinity Security Services Contributing Writer. "IDS-Can You Afford Not To Have One?". 02 Jun 2003. <http://www.networknewz.com/networknewz-10-20030602IDSCanyouaffordnottohaveone.html>
66. Trustix. "Trustix™ Personal Firewall Spyware: BackWeb / BackWeb Light Client. 2004. <http://www.personalfirewall.trustix.com/spyware/backweb.html>
67. xforce@iss.net. Fokus: ISS Alert Mailing List: ISS News Flash. 8 Feb 1999. <http://www.fokus.gmd.de/research/cc/vst/products/Security/alert/msg00071.html>
68. United States Computer Emergency Readiness Team. Vulnerability Note VU#909678: DameWare Mini Remote Control vulnerable to buffer overflow via specially crafted packets. 22 Dec. 2003 <http://www.kb.cert.org/vuls/id/909678>

Deleted: → - → -

Deleted: - -