SANS GIAC CERTIFICATION

COVER PAGE FOR ENTIRE PROJECT

Submitted by:
Jamell Creque

March 3, 2003

This paper will be a humble attempt to overview some of the pressing challenges facing Intrusion Prevention Systems, as well as Intrusion Detection systems today. The Intrusion Detection field of Network Security is still relatively new and faces a very unique challenge: how to protect valued networks and data, and at the same time, evolve to face new challenges and threats. In the past 3-4 years, advances have been made in the areas of Detection Engines – we have seen IDSes become more protocol aware and less prone to false alarms and on the other hand, event management and data correlation challenges have continued to plague the viability of every IDS vendor on the market. What are the challenges facing vendors and development teams today?

Definition of the Landscape

There are two approaches to securing computer networks in the field of Intrusion Detection: host based and network based. Host based Intrusion detection engines usually is software that resides on the valued Server or PC. It's protection tends to be based on watching protected memory spaces or looking for anomalous changes in the PC's file structure. When this activity is noticed, two actions usually occur, an alert is sent to one or more central management stations and some pre-defined level of action is taken to protect the memory block or files that are affected by the attack. Host Based Intrusion Detection tends to be very effective in stopping several types of attacks including most buffer overflows. When it is noticed that a call is made outside of the allowed memory range, the Host IDS can be very effective in the isolation and removal of the threat.

"Host-based detection systems directly monitor the host data _les and operating system processes that will potentially be targets of attack. They can, therefore, determine exactly which host resources are the targets of a particular attack." [ssnoel]

Network Based Intrusion Detection requires a standalone sensor to monitor key network segments. This sensor can be standalone or part of an array of sensor that all report to a central management station. When malicious activity is noticed, each sensor will report back to the management station what it observed and records the offending packet. Most network Based IDS systems do not respond to the offending packet for various reasons:

- The response can be too late as the offending packet has already entered the protected network. This highlights the passive nature of Network Based IDS
- If the IDS is configured to issue TCP resets to the offending source, IP spoofing can be used to deny service to legitimate computer networks

- False positive are a necessary evil with Network Based IDS and even with well tuned IDS, normal traffic can appear to be malicious and thus block service to legitimate computer networks

Network Based IDS sensors can generate thousands of alerts per day and if multiple sensors are deployed, it becomes very difficult to manage alert events.

Intrusion Prevention Systems look to protect networks by being installed in key choke points in the networks. Like Network Based IDS, IPS will monitor for malicious activity and discard offending traffic. However, IPS's protect against well known/documented Denial of Service (DoS), Distributed DoS attacks and OOS packets.

There are two main algorithms that IDS and IPS use to determine if network traffic is malicious:

- Misuse Detection – Also known as signature detection, this type of IDS/IPS compares network traffic known attack patterns. When the patterns match, and alert is generated along with the attack signature.
- Anomaly detection – An IDS/IPS is trained on a network by monitoring what is considered to be normal traffic profile. From that basis, traffic patterns can be monitored for what is considered "normal". When traffic falls significantly out of this profile, it may not necessarily be a network attack, but an alert will be generated.

Both of these detection methods have their benefits and detriments and will be examined further in the body of this paper.

State of Intrusion Detection and Prevention

Without sounding too cliché, Intrusion Detection has made huge inroads in recent years and Intrusion Prevention is still in its infancy. According to [gupta], there are 10 areas in which Intrusion Detection will continue to grow in the coming years:

- Accuracy, Accuracy, and More Accuracy
- Prevention and not just detection
- Broad Detection coverage
- Ability to capture and process all relevant traffic
- Highly Granular Detection and Response
- Granular Policy Management
- Scalable Management System
- Sophisticated Forensic Management and Reporting
- Reliable Sensor Platform

- Sensor Performance

This discourse will look to outline algorithms and methods used in achieving higher accuracy of alert events, as well as, reviewing the Intrusion Prevention model.

Accurate detection

According to Mr. Gupta, there are several areas of concern to creating more accurate Intrusion Detection Sensors. Two major

- Reduce False Positives and False Negatives

  Arguably the most plaguing aspect of intrusion detection. A False Positive is an alert generated when no malicious activity or threat exists in the packet or packets in question. A False Negative is when there is malicious or threatening activity taking place but has passed undetected by the Intrusion detection sensor. Both Misuse detection and anomaly detection engines are prone to this issue for different reasons.

  For misuse detection engines, false positives are generated when the signature alerts on traffic that is normal and non-threatening. For example: Invalid Web traffic or CGI exploits.
  False negatives occur when a new attack occurs and the sensor does not have a signature defined for that attack. Some feel that this is a major drawback to misuse detection engines. If a sensor runs with a dated signature set, it will be prone to false negative of all new attacks since the creation of its signature set.

  Anomaly detection engines are "trained" based on normal non-malicious traffic. This proves very effective when the engine has a pristine profile to compare network activity to. If the profile is not pristine, then malicious traffic will not trigger alerts producing false negatives. False positives occur when new network activity, such as a new FTP server is brought online, but the trained sensor believes that traffic to be malicious.

- Protocol Analysis & Anti-Evasion Techniques

  Signature based (misuse) detection is inherently protocol unaware, that is, it does not know or care about layers 3-7 in the OSI model. If a match cannot be made to a pre-defined string or tcp header value, false negative malicious attacks will make it through network defenses. With protocol analysis, an IDS/IPS sensor will be concerned with the payloads of the respective layers of the OSI model. For example, with protocol awareness, upper layers checksums or TTLs can be evaluated to thwart IDS evasion techniques. FTP attacks can be evaluated for normal commands versus if a back door program is running over ftp port, tcp/21. This approach can provide for more accurate and granualar IDS/IPS Systems. With more sophisticated attackers there comes a need for more sophisticated protection. IDS Evasion is the driving force behind the need for IDS/IPS systems to become more protocol aware. Smarter hackers take known good signatures and find ways to inject attacks using the very signatures against themselves.

- Accurate buffer overflow detection

Self-learning, profile driven anomaly detection

When it comes to anomaly detection, an IDS detection engine uses a profile to determine if suspected traffic is malicious in nature. The profile is generated from costly training of the IDS in known "good" network under no attack or malicious activity. This profile is quite localized and does not scale well because different segments of the network may have different traffic patterns. Each IDS sensor must be "taught" about its environment specifically. Eeskin , in a paper called adaptive model generation for Intrusion Detection Engines, addresses the challenges of anomaly IDS engines with Adaptive Model generation. This approach to Intrusion Detection allows an IDS Sensor to train itself regarding its network environment, thus building a dynamic profile. Adaptive model Generation even allows for some intrusive data in the profile training set, while still achieving a robust network profile.

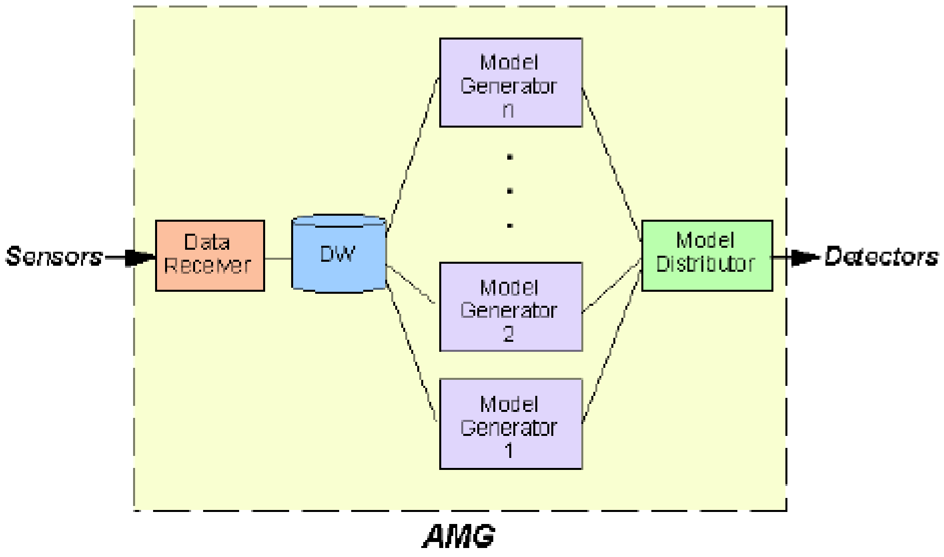


Figure 1-3 – Adaptive Model Generation

Figure 1-3[eeskin] displays the architecture for Adaptive Model Generation and is comprised of 4 elements:

1. Data receiver – receives raw data from Intrusion Sensors

2. Data Warehouse – loads the data into a database
3. Model generator runs queries on the data over time and creates data models or profiles
4. Profile Distributor – updates sensors when new models are generated

There are three elements about this design that proves valuable in the IDS/IPS arena. The first is the ability to adjust to changing network conditions and create new models or profiles on the fly. This makes the Intrusion Detection function more dynamic and therefore more effective. If the network landscape changes for the better (increased non-malicious traffic), or for the worst (active attacks), the Adaptive model can create accurate and continuous model changes.

The second issue of value with the Adaptive Model method is the ability to use various model generation algorithms in the Adaptive Model generator. [eeskin] uses a probabilistic model, but boasts that models such as heuristic or a set of rules can be used as well. The key being that the data receiver take in the training data and the Adaptive engine will create a new model set based on whatever is in the black box of the Model Generator.

The last issue of value is the ability of the model generator to create a viable model amidst noisy data. One assumption here is that, compared to the entire training data set, the attack data is a relatively small amount. With that assumption we can look at a history of system calls to a Unix machine. Normal system calls appear to be quite different than that of a malicious attack. During an attack, unauthorized system calls will be made to vulnerable programs to generate a shell with root privileges. Using a sliding window, the probabilistic model generator will try to predict the next system call given the previous history of system calls in the sliding window. When that prediction result passes a certain threshold, it declares the system call activities an intrusion. This allows the model generator to create intrusion profiles without the need for costly pristine data.

Let's consider issues of Intrusion Prevention Systems and the challenges facing creating an architecture that delivers both performance, accuracy and confidence. Intrusion Prevention systems include routers with inbound ACLs, firewalls, and dedicated sensors with robust rulesets. Together these systems protect from DoS, DDoS, Web Server and malformed/OOS attacks. Intrusion Preventions Systems today face the following challenges:

In-line Operation

In order to prevent the intrusion, the system must be in-line with the data path into the protected network. At the major bottlenecks of the network, an IPS system has to the power and ability to drop offending packets. This is very different from the IDS which was passive and, at best, was able to send TCP resets to the offending source or enable shunning ACL's on the ingress router.

Performance and negligible In-line packet processing latency

Performance is a huge issue due to the placement of IDS systems. Network Security engineers have worked hard to make firewalls keep up with network traffic demands and to eliminate the bottle neck at the entry point into the protected network. With the added load of packet reassembly and inspection, it seems that IPS can send networks steps backwards rather than forward. Multi-Gigabit speed ASICs are the answer to this question. As IDSs break the gigabit barrier, IPS is close to follow. The main key, which we will examine in the Accuracy section, is that the IPS does not have to do the entire job of the IDS. This fact will allow much needed CPU cycles to not be wasted on events that cannot be proven malicious with 99.9% accuracy.

Accuracy

According to Gupta, there is a need for unquestionable accuracy. This is actually a little easier than it sounds. The nice thing about the TCP/IP protocol is the preset rules and standards of the protocol suite. There are certain basic rules that packets should never break (dare I say!!!??).

- IP version will always be 4 or 6
- Small fragments are, more often than not, signs of bad news
- TCP reserved bits should almost never be set
- IP reserved bits should never be set
- Port 0 or ip address 0.0.0.0 should never exist
- RFC 1918 or 127.x.x.x addresses should never be seen inbound to a protected network
- Cmd.exe should never be seen in an html request
- The list goes on…

If we can apply all of these rules to the IPS system, it is very safe to say, traffic denied will be in the 99.99% probability range of being bad news. As IDS systems become more accurate, so will IPS.

Reliability and Availability (Up time)

Since IPS will be the ingress entry point into the network, hardware/software failures cannot be tolerated. Certain industries have SLA requirements of 99.99%. This is another issue where IPS can capitalize on strides made in other areas of network technology. Routers and firewalls operate in industry with the same SLA requirements and IPS will certainly benefit from that closeness.

Conclusion

Intrusion Detection and Prevention is a field that continues to grow rapidly. Driven by increasingly complex and clever attacks, inline prevention is a necessity, rather than a luxury. Until IPS systems become mature, IDS will be in the forefront, detecting and alerting with ever increasing accuracy. As IDS grows, IPS will grow along side it and both industries will drive each other to maturity.

[eeskin] - Adaptive Model Generation for Intrusion Detection Systems - Eleazar Eskin, Matthew Miller, Zhi-Da Zhong, George Yi, Wei-Ang Lee, Salvatore Stolfo. (eeskin,mmiller,zz31,georgeyi,weiang,sal]@ cs.columbia.edu

[jsnyder] – SANS Webcast – Wednesday December 4th – Intrusion Prevention Essentials by Joel Snyder, Opus One Corp.

[ssnoel] – Modern Intrusion Detection, Data Mining, and Degrees of Attack Guilt Steven Noel, Duminda Wijesekera, Charles Youman – George Mason University

[gupta] – Top 10 Requirements for next Generation IDS, Ramesh Gupta – IntruVert Networks

http://online.securityfocus.com/infocus/1544 - Protocol Analysis
http://www.security-gurus.de/papers/anomaly_rules_def.pdf - Rules based anamoly detection
http://www.sans.org/rr/intrusion/silver_bullet.php - Intrusion Prevention Systems – Security's Silver Bullet?
http://www.sans.org/rr/firewall/prevention.php - Denial of Service Attacks and the Emergence of "Intrusion Prevention Systems"

Detect 1 – Bogus fragment packet

```
07/07-08:42:38.484488 192.1.1.188 -> 46.5.244.64
TCP TTL:236 TOS:0x0 ID:0 IpLen:20 DgmLen:40 RB
Frag Offset: 0x864   Frag Size: 0x14
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+

07/07-17:35:29.094488 192.1.1.188 -> 46.5.65.28
TCP TTL:236 TOS:0x0 ID:0 IpLen:20 DgmLen:40 RB
Frag Offset: 0x864   Frag Size: 0x14
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
```

1. **Source of Trace:**

   This trace was downloaded from http://www.incidents.org/logs/Raw/index.html.

   2002.6.7 120,873 Fri Jul 19 06:13:39 2002

2. **Detect was generated by:**

   This detect was generated by Snort 1.8.7 (Build 126). I replayed the binary file into Snort output to mysql and ACID.

   According to the ACID link for the snort signature, the signature that generated this alert is as follows:

   alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"BAD TRAFFIC bad frag bits"; fragbits:MD; sid:1322; classtype:misc-activity; rev:4;)

   The packets that generate the alert is as follows:

```
08:42:38.484488 IP 192.1.1.188 > 46.5.244.64: tcp (frag 0:20@17184)
                         4500 0028 0000 8864 ec06 6770 c001 01bc
                         2e05 f440 0d40 0050 2fc7 f9fa 2fc7 f9fa
                         0004 0000 bfd1 0000 0000 0000 0000
17:35:29.094488 IP 192.1.1.188 > 46.5.65.28: tcp (frag 0:20@17184)
                         4500 0028 0000 8864 ec06 1c94 c001 01bc
                         2e05 411c 0e7d 0050 31af d116 31af d116
                         0004 0000 c1b0 0000 0000 0000 0000
```

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

   The probability that the source address was spoofed is high due to the fact that the packet is intended to DoS the target IP address. Also, the attacker does not need a response from the target IP address in order for this attack to be successful

4. Description of attack:

IP reserved bit is set – This bit is not used in the TCP/IP protocol. There is not a benign reason for this bit to be set. See Figure 1.

IPID = 0 – This value does not occur in the natural TCP/IP protocol stack.

Fragment offset value is bogus. No other packet existed in the trace that this fragment belonged to. Even if such a packet train existed, it would be suspect due to the IP ID being 0.

In my humble opinion, this packet is up to no good. This is either an OS fingerprint packet (albeit a poorly designed one) or a DoS packet targeted towards MS Windows workstations and servers.

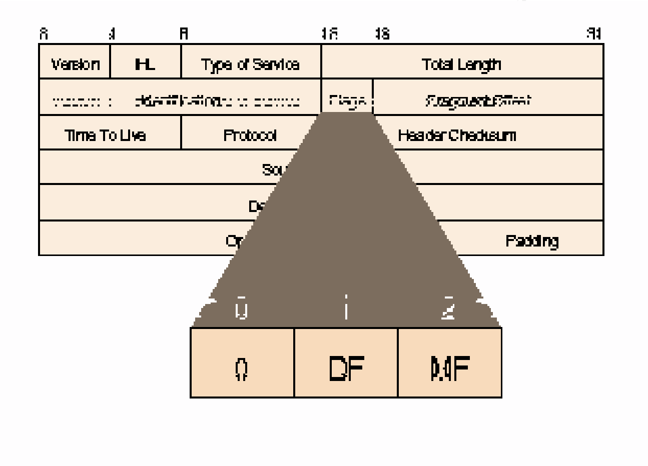


Figure 1

5. Attack mechanism:

For unpatched MS Windows servers and workstations, 100-150 of these malformed packets per second can cause the operating system to hang. An unpatched Windows TCP/IP protocol stack will try to reassemble this fragment and in the process utilize 100% of its CPU. In some cases the machine can recover when the stream of packets subside

6. Correlations:

The following resources were used to gather information and correlate findings on “Reserved bit set”, “Fragment offset” and “IPID=0”:

http://www.snort.org/snort-db/sid.html?id=523
http://archives.neohapsis.com/archives/snort/2001-10/0357.html
http://www.ntbugtraq.com/default.asp?pid=36&sid=1&A2=ind0005&L=ntbugtraq&F=&S=&P=10991
Microsoft's security bulletin:
http://www.microsoft.com/technet/security/bulletin/ms00-029.asp

CAN2000-0305

7. **Evidence of active targeting:**

The only evidence of active targeting is purely inferred. This attack is a DoS attempt directed towards unpatched Windows NT desktops or servers. The attacker would have performed reconnaissance to determine that this type of host exists and directs this attack in hopes of interrupting the systems normal operations.

8. **Severity:**

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

Criticality: 4 (Windows Server)

Lethality: 3 (Could bring down the server if unpatched, however only one packet per destination is not enough to crash the server)

System countermeasures: 5 (modern OS, patches applied)

Network countermeasures: 1 (fragments are allowed into the network)

Severity = (3 + 4) – (5 + 1) = **+1**

9. **Defensive recommendation:**

Two defensive recommendations for this attack:

1 - Windows NT 4.0 Server, Terminal Server Edition:
http://www.microsoft.com/Downloads/Release.asp?ReleaseID=20830

Windows 2000 Professional, Server and Advanced Server:
http://www.microsoft.com/Downloads/Release.asp?ReleaseID=20827

2 – Filter fragmented packets at the ingress router

10. **Multiple choice test question:**

What is the IPID of the following fragmented packet?

```
08:42:38.484488 IP 192.1.1.188 > 46.5.244.64: tcp (frag 0:20@17184)
                          4500 0028 0000 8864 ec06 6770 c001 01bc
                          2e05 f440 0d40 0050 2fc7 f9fa 2fc7 f9fa
                          0004 0000 bfd1 0000 0000 0000 0000
```

a. 1260
b. 1261
c. 0x0000
d. 0x28

Answer: C

Detect 2 – IIS Unicode Detect

1. **Source of Trace:**

This trace was detected on the perimeter of my company network. The SNORT IDS sensor sits in front of the firewall but behind the perimeter router. The firewall is Cisco PIX 515 and the Router is a Cisco 3640.

2. **Detect was generated by:**

This detect was generated by Snort 1.8.7 (Build 126)

The alert that was generated is as follows:

```
1/25-10:08:45.162231  [**] SPP_HTTP_DECODE: IIS UNICODE ATTACK
DETECTED [**] 9.67.214.111:2047 -> X.X.201.54:80
```

9.67.214.111 is the outside network and X.X.201.54 is the private network.

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

The probability that the source address was spoofed is low. This is an access attack and the attacker wants to get information off the computer so the source address has to be an active machine that the attacker has control of.

4. **Description of attack:**

If this attack is successful, unauthenticated users can retrieve any file in relation to the privileges of the IUSR account. An attacker on a website hosted by IIS, can exploit a vulnerability in IIS 4.0 and 5.0 called Web Server Folder Directory Transversal.

5. **Attack mechanism:**

Any url similar to:

http://www.yourweb.net/../../../winnt/calc.exe

can launch and run any program.

The following is a list of vulnerable operating systems:

Microsoft IIS 4.0 alpha
- Microsoft Windows NT 4.0 alpha

Microsoft IIS 4.0
+ Cisco Building Broadband Service Manager 5.0
+ Cisco Call Manager 1.0
+ Cisco Call Manager 2.0
+ Cisco Call Manager 3.0
+ Cisco ICS 7750
+ Cisco IP/VC 3540
+ Cisco Unity Server 2.0
+ Cisco Unity Server 2.2
+ Cisco Unity Server 2.3
+ Cisco Unity Server 2.4
+ Cisco uOne 1.0
+ Cisco uOne 2.0
+ Cisco uOne 3.0
+ Cisco uOne 4.0
+ Microsoft BackOffice 4.0
+ Microsoft BackOffice 4.5
+ Microsoft Windows NT 4.0 Option Pack

Microsoft IIS 5.0
+ Microsoft Windows 2000 Advanced Server
- Microsoft Windows 2000 Advanced Server SP1
- Microsoft Windows 2000 Advanced Server SP2
- Microsoft Windows 2000 Datacenter Server SP1
- Microsoft Windows 2000 Datacenter Server SP2
+ Microsoft Windows 2000 Professional
- Microsoft Windows 2000 Professional SP1
- Microsoft Windows 2000 Professional SP2
+ Microsoft Windows 2000 Server
- Microsoft Windows 2000 Server SP1
- Microsoft Windows 2000 Server SP2

Microsoft Personal Web Server 4.0
+ Microsoft NT Option Pack for NT 4.0
+ Microsoft Windows 98

6. **Correlations:**

   http://www.securityfocus.com/frames/?content=/vdb/bottom.html%3Fvid%3D1806

   http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS00-078.asp - FAQ section

7. **Evidence of active targeting:**

   There was no evidence of active targeting. Our organization was scanned 10,000+ times for this from 2-17-2003 to 3-2-2003. This is a very common scan.

8. **Severity:**

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

   Criticality: 5 (Windows Web Server)

   Lethality: 5 (Attacker can access files and execute programs)

   System countermeasures: 5 (OS patches applied)

   Network countermeasures: 4 (WebServer is in Firewall DMZ and port 80 traffic is only allowed to one destination address due to ACLs on PIX firewall)

   Severity = (5 + 5) – (5 + 4) = **+1**

   This level of severity is an acceptable risk because our organization must have a web server.

9. **Defensive recommendation:**

   Load the following patch from Microsoft:

   http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS00-078.asp

10. **Multiple choice test question:**

What user account’s permissions does the Web Server Folder Directory Transversal attack utilize for its attack on a Web Server named CORPDC03?

A. cmd.exe
B. CORPDC03
C. IUSR_CORPDC03
D. IWAM_CORPDC03

Answer: C

Detect 3 – SMB Name WildCARD SCAN

1. **Source of Trace:**

Unknown – The assumption is that this network is open to the internet and does not have an ingress packet filtering policy.

2. **Detect was generated by:**

Unknown – this detect was taken from:

http://lists.insecure.org/lists/incidents/2000/Apr/0047.html

```
Apr 12 03:20:10 host snort: SMB Name Wildcard:
209.112.188.221:137 -> x.x.x.99:137
Apr 12 03:20:10 host snort: SMB Name Wildcard:
169.254.222.20:137 -> x.x.x.99:137
Apr 12 03:20:10 host snort: SMB Name Wildcard:
209.112.188.221:137 -> x.x.x.99:137
Apr 12 03:20:12 host snort: SMB Name Wildcard:
169.254.222.20:137 -> x.x.x.99:137
Apr 12 03:20:12 host snort: SMB Name Wildcard:
209.112.188.221:137 -> x.x.x.99:137
Apr 12 03:21:17 host snort: SMB Name Wildcard:
209.112.188.221:137 -> x.x.x.104:137
Apr 12 03:21:18 host snort: SMB Name Wildcard:
169.254.222.20:137 -> x.x.x.104:137
Apr 12 03:21:18 host snort: SMB Name Wildcard:
209.112.188.221:137 -> x.x.x.104:137
Apr 12 03:21:20 host snort: SMB Name Wildcard:
169.254.222.20:137 -> x.x.x.104:137
Apr 12 03:21:20 host snort: SMB Name Wildcard:
209.112.188.221:137 -> x.x.x.104:137
Apr 12 03:23:02 host snort: SMB Name Wildcard:
209.112.188.221:137 -> x.x.x.112:137
Apr 12 03:23:44 host snort: SMB Name Wildcard:
169.254.222.20:137 -> x.x.x.115:137
Apr 12 03:26:03 host snort: SMB Name Wildcard:
209.112.188.221:137 -> x.x.x.122:137
```

```
Apr 12 03:27:15 host snort: SMB Name Wildcard:
209.112.188.221:137 -> x.x.x.124:137
Apr 12 03:27:16 host snort: SMB Name Wildcard:
169.254.222.20:137 -> x.x.x.124:137
Apr 12 03:27:16 host snort: SMB Name Wildcard:
209.112.188.221:137 -> x.x.x.124:137
Apr 12 03:27:18 host snort: SMB Name Wildcard:
169.254.222.20:137 -> x.x.x.124:137
Apr 12 03:27:18 host snort: SMB Name Wildcard:
209.112.188.221:137 -> x.x.x.124:137
Apr 12 03:27:40 host snort: SMB Name Wildcard:
209.112.188.221:137 -> x.x.x.126:137
```

3. Probability the source address was spoofed:

The probability that the source address was spoofed is high. Although this is a reconnaissance attack and the attacker needs to know if the scanned computer replied, it appears that 2 different sources generate the scans.

Description of attack:

Microsoft Operating Systems that respond to NETBIOS name retrieval are vulnerable. An attack trace would look like:

```
12/30-02:28:32.282973 source:1057 -> target:137
UDP TTL:64 TOS:0x0 ID:62089
Len: 58
24 C0 00 00 00 01 00 00 00 00 00 00 20 43 4B 41  $........... CKA
41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41  AAAAAAAAAAAAAAAA
41 41 41 41 41 41 41 41 41 41 41 41 41 00 00 21  AAAAAAAAAAAAA..!
00 01                                            ..
```

4. Attack mechanism:

The attack is from spoofed addresses and is probably from an automated scanning tool. Unless the attacker has control of both source machines, this attack will not be very successful. The goal of the attack is to gain the following information:

- Windows Domain Name
- Usernames of users or administrators currently logged on
- NETBIOS name of the workstation or server

5. Correlations:

http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0621

http://www.whitehats.com/info/IDS177

6. **Evidence of active targeting:**

   There was no evidence of active targeting. This appears to be a slow scan but a scan nonetheless. This is a very common scan.

7. **Severity:**

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

   Criticality: 4 (Very simple reconnaissance exploit)

   Lethality: 4 (Target systems can be easily exploited once identified)

   System countermeasures: 2 (MS OS allows NETBIOS communication by default)

   Network countermeasures: 2 (WebServer is in Firewall DMZ and port 80 traffic is only allowed to one destination address due to ACLs on PIX firewall)

   Severity = (4+ 4) – (2+ 2) = **-4**

   This level of severity is not an acceptable risk because the target network's exposure is dependant upon the attaker not having control of the source machines.

8. **Defensive recommendation:**

   Filter port 137 on the ingress router.

9. **Multiple choice test question:**

   What is the most effective defense against SMB Name Wildcard scans?

   A. Use WINS instead
   B. Allow only authenticated NETBIOS connections
   C. Filter port 137 at the perimeter
   D. Install Active Agent Directory

   Answer: C

Responses to Detect #1
From Brian Coyle - brian@linuxwindows.com

```
> The signature that generated this alert is as follows:
> alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"BAD TRAFFIC bad frag
> bits"; fragbits:MD;
> sid:1322; classtype:misc-activity; rev:4;)
```

What in particular in the captured packets caused this rule to fire? (see also #6 below - this might be the wrong rule).

Well – This is hard to explain. I ran the binary files through my instance of SNORT and ACID. When I clicked the link on the alert it lead me to the above alert. I ASSUMED it was correct but it is clearly not. The packet in tcpdump output shows the reserved bit set only. This rule triggers on the More Fragment and Don’t Fragment bit. So – my overall answer is that I don’t know why this rule fired but I swear it did.

> IPID = 0 ? This value does not occur in the natural TCP/IP protocol stack.

Huh? I've got lots of packets with IPID==0. I suggest a review of Stevens and RFC791 to clarify your statement.

I am obviously incorrect. I searched seven websites looking to see if IPID=0 is possible. I can clearly see a programmer somewhere programming correctly and starting with the true 1st number. However, I looked and could not find the correlation and I forged ahead.

From: **Andrew Rucker Jones <arjones@simultan.dyndns.org>**

> For unpatched MS Windows servers and workstations,
> 100-150 of these malformed packets per second can
> cause the operating system to hang. An unpatched
> Windows TCP/IP protocol stack will try to reassemble
> this fragment and in the process utilize 100% of its CPU.
> In some cases the machine can recover when the stream
> of packets subside

I think this is evidence that this is NOT a DoS attack. If someone had attempted that, You would likely have seen at least a few more packets per host, don't You think?

I really researched this and saw the potential for this to lock a Windows machine. I could lower my Lethality or abandon all hope od a DoS and start over. Next time, if faced with the same two packets, I probably would go for a false positive rather than a DoS attack or OS fingerprint.

I - Executive Summary

This report provides a security audit of an enterprise using data analysis. Data has been collected over 5 consecutive days and is listed in Table 1. Analysis of the data consisted of various utilities and software programs. The overall goal is to provide a thorough analysis of security threats, vulnerabilities, and intrusions. The results of this analysis should be carefully considered and action should be taken based on the severity of various events. Section II discusses the files examined in detail. Section III is the analysis section and begins with a list of the "top talkers" and the most prolific traffic patterns that were observed. Additional analyses performed in this section will illustrate some very obvious, as well as obscure, security events that affect the enterprise. Section IV lists the systems in the network and their ranking in terms of likely guilt and probable guilt as it relates to the overall security of the enterprise. This section also includes recommendations regarding countermeasures that can be deployed to better protect the enterprise. Finally, section V provides a brief overview of the methods and applications used to analyze the data.

Overall findings:

The enterprise network exhibited serious security issues in the following areas:

- Windows and Webserver Security
- Many enterprise machines appear to have been co-opted and some are being used to attack external machines
- Several Trojan servers and DDoS agents appear to be active on the network

On 2/12 and 2/13 major DDoS attack occurred on 2 external addresses

Conclusions

Although serious security vulnerabilities exist in the enterprise network, with diligence and followup verification, all serious issues can be corrected. This type of analysis should be conducted on the enterprise network monthly in order to ensure network viability and to mitigate further security intrusions.

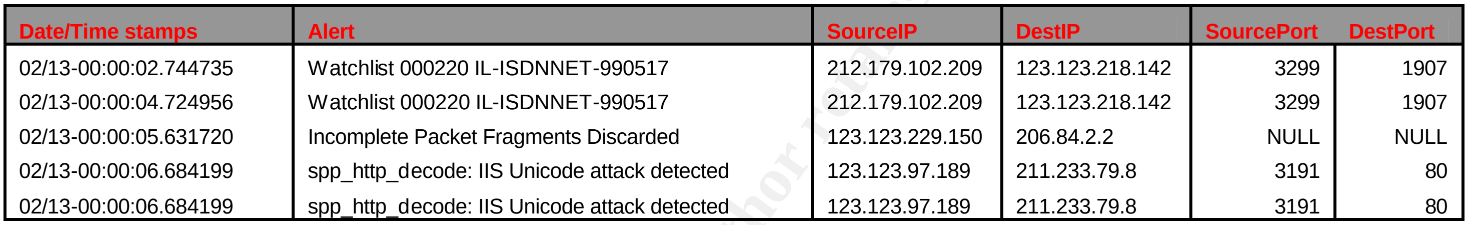
II - Data files used in this analysis

Alerts	Scans	Out-of-Spec Files
alert.030212.gz	scans. 030212.gz	oos_2003_02_12.gz
alert.030213.gz	scans. 030213.gz	oos_2003_02_13.gz
alert.030214.gz	scans. 030214.gz	oos_2003_02_14.gz
alert.030215.gz	scans. 030215.gz	oos_2003_02_15.gz
alert.030216.gz	scans. 030216.gz	oos_2003_02_16.gz

Figure 1: Data files used for security audit from Feb 12 2003 to Feb 12, 2003.

The format of these files are as follows:

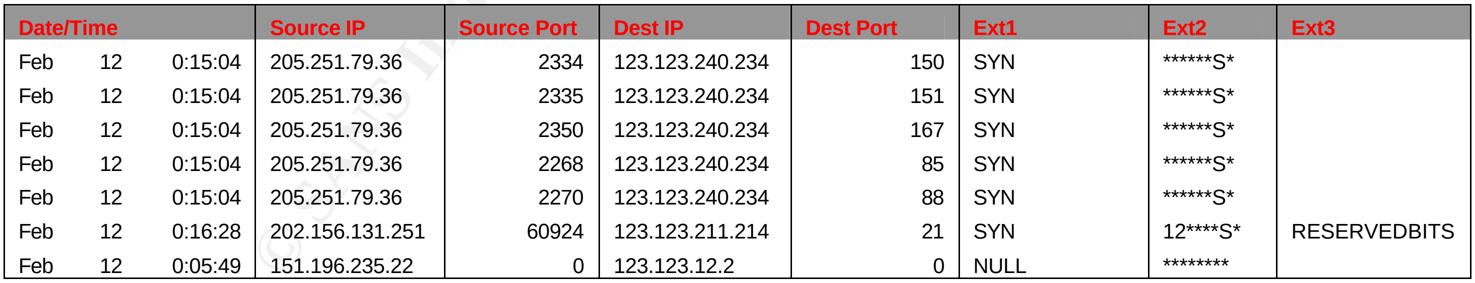
alert files:



Date/Time stamps	Alert	SourceIP	DestIP	SourcePort	DestPort
02/13-00:00:02.744735	Watchlist 000220 IL-ISDNNET-990517	212.179.102.209	123.123.218.142	3299	1907
02/13-00:00:04.724956	Watchlist 000220 IL-ISDNNET-990517	212.179.102.209	123.123.218.142	3299	1907
02/13-00:00:05.631720	Incomplete Packet Fragments Discarded	123.123.229.150	206.84.2.2	NULL	NULL
02/13-00:00:06.684199	spp_http_decode: IIS Unicode attack detected	123.123.97.189	211.233.79.8	3191	80
02/13-00:00:06.684199	spp_http_decode: IIS Unicode attack detected	123.123.97.189	211.233.79.8	3191	80

Figure 2 – Alert file example of alerts

scan files:



Date/Time			Source IP	Source Port	Dest IP	Dest Port	Ext1	Ext2	Ext3
Feb	12	0:15:04	205.251.79.36	2334	123.123.240.234	150	SYN	******S*	
Feb	12	0:15:04	205.251.79.36	2335	123.123.240.234	151	SYN	******S*	
Feb	12	0:15:04	205.251.79.36	2350	123.123.240.234	167	SYN	******S*	
Feb	12	0:15:04	205.251.79.36	2268	123.123.240.234	85	SYN	******S*	
Feb	12	0:15:04	205.251.79.36	2270	123.123.240.234	88	SYN	******S*	
Feb	12	0:16:28	202.156.131.251	60924	123.123.211.214	21	SYN	12****S*	RESERVEDBITS
Feb	12	0:05:49	151.196.235.22	0	123.123.12.2	0	NULL	********	

Figure 3 – Scan file example of alerts

OOS files:

```
=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
=+=+=+=+=+

02/12-00:07:46.734338 80.13.189.98:41355 -> MY.NET.202.50:6346
TCP TTL:45 TOS:0x0 ID:24997 IpLen:20 DgmLen:60 DF
12****S* Seq: 0x4BC930F7  Ack: 0x0  Win: 0x16D0  TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 32789119 0 NOP WS: 0

=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+=+
=+=+=+=+=+

02/12-00:10:10.910912 202.156.131.251:60719 -> MY.NET.211.214:21
TCP TTL:45 TOS:0x0 ID:42446 IpLen:20 DgmLen:60 DF
12****S* Seq: 0x7BB2C0D7  Ack: 0x0  Win: 0x16D0  TcpLen: 40
TCP Options (6) => MSS: 1460 NOP NOP TS: 336488006 0 NOP WS: 0
```

The alert files contained portscan information and that data was analyzed in the scan files. 2/12, 2/15, and 2/16 were files of similar size and contained mostly the same information. One of the first anomalies noticed however, was the sizes of 2/13 and 2/14 alert files. Respectively these files were 155MB and 74MB. Section III will address reasons for the size differences between files on the dates in question.

The private network was obfuscated with MY.NET for the first 2 octets of the address and every instance of MY.NET was replaced with 123.123. Thus, 123.123 addresses represent the private enterprise network addresses. When importing the data, a small amount of data corruption was present in the alert files. Upon closer inspection, it appears that the IDS hard disk was not able to keep up with the write function. This occurred 6621 times and an example of the corrupted data follows:

The corrupted data amounted to 0.42% of the total alerts and did not impact the overall analysis.

III - Analysis

A - Top Ten Analysis

The alerts data set contained 1,585,204 events. There existed 38,421 scan events and 6621 corrupted events. The following is an analysis of the remaining 1,509,362 events

Figure 4, 5, and 6 is a representation of the top talkers for Source IP address, Destination Port, and Alerts respectively. This data set is based on aggregate totals of occurrences.

Top 10 Talkers by Source IP	# of Occurrences
NULL	45077
123.123.229.150	7168
212.179.123.163	2564
141.157.254.236	2238
212.179.107.228	1742
205.251.79.36	1683
128.54.135.205	1556
123.123.85.74	1491
212.179.75.145	1454
212.179.35.118	1416

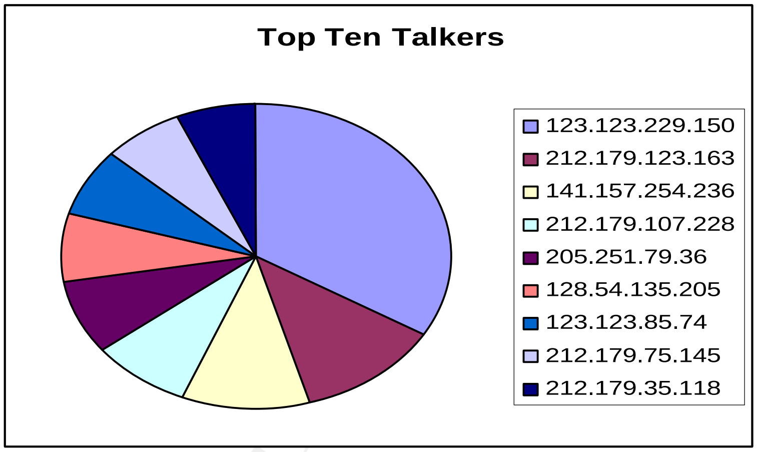


Figure 4 – Top talkers by IP address

The top 10 talkers by source IP data set was very interesting for a few reasons:

- 4 of the top ten talkers appear to be from the same network – 212.179 – class B
- The top talker is internal to the network is port scans – An alert with a NULL source and Destination IP address is attributed to the spp_portscan alert. This alert is an indicator that port scanning is taking place.
- The #2 and #8 top talkers are internal addresses

Top 10 Talkers by Dest IP	# of Occurrences
209.126.247.144	936136
203.198.175.211	467694
NULL	45077
123.123.100.165	9856
206.84.2.2	4082
209.133.9.106	3108
123.123.235.62	2677
123.123.252.126	1783
192.168.0.253	1414
123.123.224.18	1182

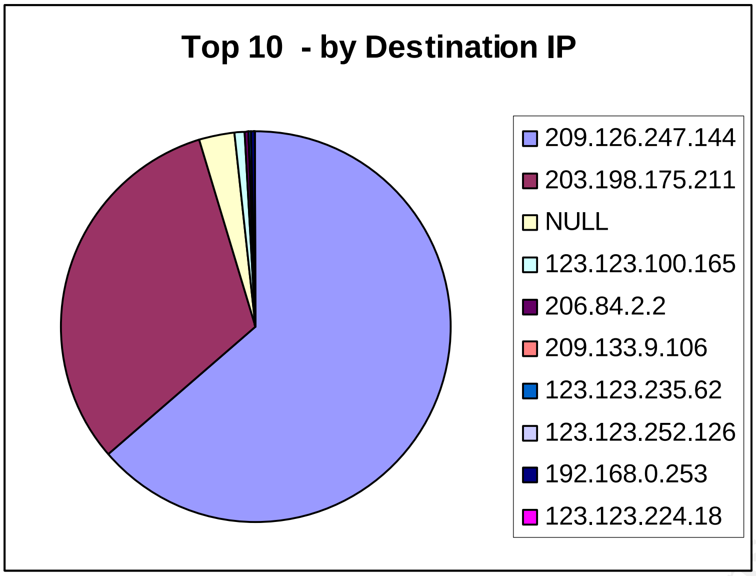


Figure 5 – Top talkers by destination IP address

The top 10 talkers by destination address resulted in some raised eyebrows. Immediately it is noticed that the top 2 addresses comprise 93% of the alerts for all 5 days. Upon further inspection these alerts actually take place on 2/13 and 2/14. Other issues of interest:

- 192.168.0.253 is a RFC 1918 and should not be routed to the enterprise network
- 4 destination addresses are not on the enterprise network. It is expected that hostile activity is direct toward the enterprise network. These 4 addresses represent malicious traffic direct out of the enterprise network.

Top 10 Dest Ports	# of Occurrences
137	80185
NULL	52470
80	25951
65535	4597
1321	2543
32771	2408
0	1533
3323	1203
55850	1193
1214	1126
1210	995

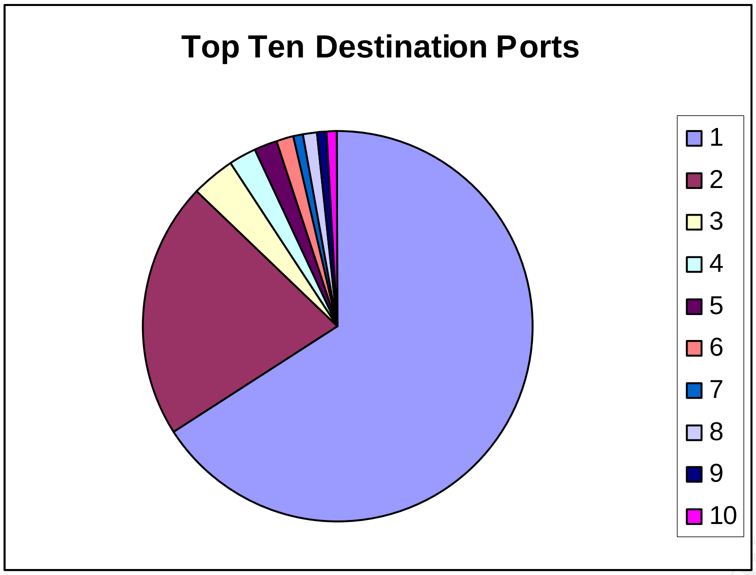


Figure 6 – Top 10 destination ports

In Figure 3, listed are the top 10 destination ports on the enterprise. Some ports of interest are as follows:

- UDP 137 – Windows NETBIOS communication port
- TCP 65535 – RC1 Trojan or RedWorm
- TCP 0 – Rex
- TCP 1321 – No known Trojan or exploit
- TCP 32771 – Remote Procedure Call – RPC
- TCP 55850 – myserver DDoS
- 1214 – KaaZa file sharing
- 1210 - No known Trojan or exploit

Here is a total listing of every alert present on the network, not including portscan alerts and alerts with NULL for the signature.

B – Signature Analysis

Alerts	# of Occurances
TCP SRC and DST outside network	1403842
SMB Name Wildcard	80163
Watchlist 000220 IL-ISDNNET-990517	15759
spp_http_decode: IIS Unicode attack detected	11342
CS WEBSERVER - external web traffic	9574
Incomplete Packet Fragments Discarded	7788
High port 65535 tcp - possible Red Worm - traffic	6752
spp_http_decode: CGI Null Byte attack detected	3381
SUNRPC highport access!	2346
TFTP - Internal TCP connection to external tftp server	1730
Port 55850 tcp - Possible myserver activity - ref. 010313-1	1694
TFTP - External UDP connection to internal tftp server	1413
Null scan!	1080
High port 65535 udp - possible Red Worm - traffic	1042
Possible trojan server activity	994
Watchlist 000222 NET-NCFC	943
IDS552/web-iis_IIS ISAPI Overflow ida nosize	797
123.123.30.4 activity	726
Queso fingerprint	700
EXPLOIT x86 NOOP	577
123.123.30.3 activity	439
NETBIOS NT NULL session	280
CS WEBSERVER - external ftp traffic	187
IRC evil - running XDCC	155
NMAP TCP ping!	137
Tiny Fragments - Possible Hostile Activity	114
connect to 515 from outside	81
EXPLOIT x86 setuid 0	67
EXPLOIT x86 stealth noop	54
EXPLOIT x86 setgid 0	39
Attempted Sun RPC high port access	36
SNMP public access	36
TFTP - Internal UDP connection to external tftp server	25
IDS552/web-iis_IIS ISAPI Overflow ida INTERNAL nosize	23
TFTP - External TCP connection to internal tftp server	20
DDOS mstream handler to client	14
Notify Brian B. 3.56 tcp	9
SMB C access	9
RFB - Possible WinVNC - 010708-1	8
External RPC call	7
FTP passwd attempt	7
Notify Brian B. 3.54 tcp	6
Port 55850 udp - Possible myserver activity - ref. 010313-1	5
DDOS mstream client to handler	2
Bugbear@MM virus in SMTP	1

Figure 7 – Aggregate listing of all alerts

A significant amount of time is spent analyzing most of these alerts and although the alerts with the greatest aggregate totals are important, a good analysis will focus on the smaller attacks in the hopes of finding the method of gaining control of target machines. Some of the smaller alerts of interest are as follows:

- DDOS mstream client to handler
- DDOS mstream handler to client
- Tiny Fragments - Possible Hostile Activity
- Bugbear@MM virus in SMTP
- connect to 515 from outside
- RFB - Possible WinVNC - 010708-1
- FTP passwd attempt
- NETBIOS NT NULL session
- TFTP - External TCP connection to internal tftp server

Although these aggregate totals gave a good beginning for the analysis of the enterprise, more questions than answers began to arise. Below is a top talkers list based on unique promiscuous occurrence, that is, how many times a source IP or signature appeared in the data to multiple unique destinations. For example, in the total aggregate occurrences, Source IP addresses 209.126.247.144 and 203.198.175.211 triggered the signature – TCP SRC and DST outside network - 936136 and 467694 times respectively. These are very clearly the top talkers in regards to destination addresses, however, these addresses do not appear in the data in any other instances. Below, in Figure 5, are listings of all sources that have unique promiscuous occurrences, that is, all source IP addresses are referenced in relation to how many unique connections are made and to unique destinations. In the unique occurrences table, an address of interest has a high number of connections to multiple sources/destinations. For example, source IP address 12.35.1158.199 connected or attemted to connect to 1,013 unique destinations. Both top talkers and top unique destinations are of extreme concern and I make this distinction to try to find the most pervasive security threats that exist in the enterprise network.

Source IP address	Unique Destinations
12.35.158.199	1013
205.251.79.36	976
65.40.3.166	325
207.6.57.6	209
67.83.29.116	137

Figure 8 – Unique Occurrences Table

C - ARIN Lookups

2 victims
209.84.2.2 – Destination IP that was attacked by 123.123.229.150

```
OrgName:    AGIS
OrgID:      AGIS
Address:    P.O. Box 9268
City:       Reston
StateProv:  VA
PostalCode: 20195-3168
Country:    US

NetRange:   206.84.0.0 - 206.85.255.255
CIDR:       206.84.0.0/15
NetName:    ALERON-206-84
NetHandle:  NET-206-84-0-0-1
Parent:     NET-206-0-0-0-0
NetType:    Direct Allocation
NameServer: NS1.ALERON.NET
NameServer: NS2.ALERON.NET
NameServer: NS3.ALERON.NET
NameServer: NS4.ALERON.NET
Comment:    This block is non-portable
RegDate:    1995-08-10
Updated:    2002-09-12

TechHandle: ADA2-ARIN
TechName:   Administrator, Aleron DNS
TechPhone:  +1-703-375-5600
TechEmail:  dns-admin@aleron.net

OrgAbuseHandle: ALERO-ARIN
OrgAbuseName:   Aleron Abuse
OrgAbusePhone:  +1-703-375-5600
OrgAbuseEmail:  abuse@aleron.com

OrgTechHandle: ADA2-ARIN
OrgTechName:   Administrator, Aleron DNS
OrgTechPhone:  +1-703-375-5600
OrgTechEmail:  dns-admin@aleron.net
```

209.126.247.144 – Destination Address involved in TCP SRC and DST outside network alert.

```
OrgName:    California Regional Internet, Inc.
OrgID:      CALI
Address:    8929A COMPLEX DRIVE
City:       SAN DIEGO
StateProv:  CA
PostalCode: 92123
Country:    US

NetRange:   209.126.128.0 - 209.126.255.255
CIDR:       209.126.128.0/17
NetName:    CARI
NetHandle:  NET-209-126-128-0-1
Parent:     NET-209-0-0-0-0
NetType:    Direct Allocation
NameServer: NS1.ASPADMIN.COM
NameServer: NS2.ASPADMIN.COM
Comment:    ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE
RegDate:    1999-03-12
Updated:    2002-06-03

TechHandle: IC63-ARIN
TechName:   California Regional Intranet, Inc.
TechPhone:  +1-858-974-5080
TechEmail:  sysadmin@cari.net
```

Top 3 Attackers ARIN Lookups

212.179.123.163 – According to Jason Lam, GCIA, this address is part of the ripe network who's whois lookup on ripe.net returns ISDN network of Isreal.

```
OrgName:    RIPE Network Coordination Centre
OrgID:      RIPE
Address:    Singel 258
Address:    1016 AB
City:       Amsterdam
StateProv:
PostalCode:
Country:    NL

NetRange:   212.0.0.0 - 212.255.255.255
CIDR:       212.0.0.0/8
NetName:    RIPE-NCC-212
NetHandle:  NET-212-0-0-0-1
Parent:
NetType:    Allocated to RIPE NCC
NameServer: NS.RIPE.NET
```

```
NameServer: AUTH03.NS.UU.NET
NameServer: NS2.NIC.FR
NameServer: SUNIC.SUNET.SE
NameServer: MUNNARI.OZ.AU
NameServer: NS.APNIC.NET
Comment:    These addresses have been further assigned to users in
Comment:    the RIPE NCC region. Contact information can be found in
Comment:    the RIPE database at whois.ripe.net
Comment:
RegDate:    1997-11-14
Updated:    2002-09-11

OrgTechHandle: RIPE-NCC-ARIN
OrgTechName:   RIPE NCC Hostmaster
OrgTechPhone:  +31 20 535 4444
OrgTechEmail:  nicdb@ripe.net
```

141.157.254.236

```
OrgName:    Verizon Internet Services
OrgID:      VRIS
Address:    1880 Campus Commons Dr
City:       Reston
StateProv:  VA
PostalCode: 20191
Country:    US

NetRange:   141.149.0.0 - 141.158.255.255
CIDR:       141.149.0.0/16, 141.150.0.0/15, 141.152.0.0/14, 141.156.0.0/15,
141.158.0.0/16
NetName:    VIS-141-149
NetHandle:  NET-141-149-0-0-1
Parent:     NET-141-0-0-0-0
NetType:    Direct Allocation
NameServer: NSDC.BA-DSG.NET
NameServer: GTEPH.BA-DSG.NET
Comment:
RegDate:
Updated:    2002-08-22
TechHandle: ZV20-ARIN
TechName:   Verizon Internet Services
TechPhone:  +1-703-295-4583
```

```
TechEmail:  noc@gnilink.net

OrgTechHandle: ZV20-ARIN
OrgTechName:   Verizon Internet Services
OrgTechPhone:  +1-703-295-4583
OrgTechEmail:  noc@gnilink.net

OrgAbuseHandle: VISAB-ARIN
OrgAbuseName:   VIS Abuse
OrgAbusePhone:  +1-703-295-4583
OrgAbuseEmail:  abuse@verizon.net



205.251.79.36

OrgName:    Cable Atlantic Inc.
OrgID:      CBAT
Address:    22 Austin Street
City:       St. John's
StateProv:  NL
PostalCode: A1B-3P2
Country:    CA

NetRange:   205.251.0.0 - 205.251.255.255
CIDR:       205.251.0.0/16
NetName:    CABLEATLANTIC-3
NetHandle:  NET-205-251-0-0-1
Parent:     NET-205-0-0-0-0
NetType:    Direct Allocation
NameServer: DNS.NF.NET
NameServer: DNS2.NF.NET
Comment:
RegDate:    2000-11-06
Updated:    2000-11-06

TechHandle: DA3001-ORG-ARIN
TechName:   DNS Administrator
TechPhone:  +1-709-753-7583
TechEmail:  dnsadmin@thezone.net

OrgAbuseHandle: RAN20-ARIN
OrgAbuseName:   Rogers Abuse - Newfoundland
OrgAbusePhone:  +1-709-753-7583
OrgAbuseEmail:  abuse@rogers.nf.net

OrgTechHandle: DA3001-ORG-ARIN
```

OrgTechName: DNS Administrator
OrgTechPhone: +1-709-753-7583
OrgTechEmail: dnsadmin@thezone.net

D – Analysis and Correlation

Scan Analysis

Looking deeper into the scan data files, the following were the top 10 scanners.

Source IP address	Scan Type	# of Occurrences	Notes
123.123.223.78	SYN	111366	443 and 80
123.123.70.176	UDP	20482	6257 and 4 packets to UDP 65535
123.123.87.44	UDP	13050	27005 and various destports
212.171.55.114	SYN	7231	All TCP 8888 - Napster? - 137 & 80 inbound
123.123.82.239	SYN	7089	Mostly 135 – some to ports 139, 443, 445
66.134.226.37	SYN	5380	443 and 80
123.123.97.110	UDP	4198	137 and 139
123.123.97.67	UDP	3799	7674 and 22321
123.123.252.82	SYN & UDP	2814	SYN to TCP 445 - UDP to various
123.123.97.35	UDP	2794	7674 and 22321

Figure 9 – Scan File Analysis by # of Occurrences

- There are well documented attacks on all of the following ports: 80, 443,4are well documented attacks on all of the following ports: 80, 443,445, 135, 137, 139
- Of the remaining ports, there is evidence of scans occurring to ports 7674 and 22321 and port 8888 has been used for Naptster and Dark IRC Trojan

Denial of Service attacks

From the alert and scan data, there appears to be a lot of activity connecting to 2 known DDoS server ports: 55850 and 6346 which are myserver and mstream DDoS agents respectively. Roland Lee, GCIA, talks about msteam DDoS in his practical. Link Graph item 3 and 4 illustrates outbound connections to 55850. Notice the numerous inbound connections inbound on port 137 (SMB Name Wildcard) and from the “Watchlist” IP addresses on port 137. The SMB Name Wildcard scan is an attempt to get the following information from a workstation or server:

- Windows Domain Name
- Usernames of users or administrators
- NETBIOS name of the workstation or server

Loras Even, GCIA, and Jason Lam, GCIA, talk about SMB Name Wildcard scans in their practical. This suspicious traffic occurs before the traffic is noticed being sent to destination port 55850. The results of which strongly suggests that co-opted machines on Windows ports, 137-139, and 445, could have been utilized to launch DoS attacks against the following destination IP addresses:

209.126.247.144	– TCP SRC and DST outside network – 936,136 alerts
203.198.175.211	– TCP SRC and DST outside network – 467, 694 alerts
206.84.2.2	– Incomplete Packet Fragments Discarded – 4082 alerts
209.133.9.106	– Incomplete Packet Fragments Discarded – 3108 alerts

From the alert and scan data, there appears to be a lot of activity to 2 known Trojan ports: 65535 and 27374 which are Red Worm/RC1 Trojan and SubSeven Trojan respectively. The presence of these alerts indicates that many machines may have been compromised. www.whitehats.com com states, “Most commonly these trojans are limited “remote administration tools” that allow an attacker to take complete control over the victim server.” This further supports the notion that internal machines are being used to attack or co-opt machines outside of the enterprise.

OOS top talkers

Source IP address	# of Occurences
148.63.130.172	401
202.156.131.251	305
148.64.169.5	286
80.222.91.197	216
212.73.96.111	190
68.164.35.154	190
210.253.215.113	173
200.163.200.5	155
209.104.74.2	144
61.114.222.241	143

Figure 10 – OOS File Analysis by Number of Occurrences

E – LinkGraph and Analysis

Figure 11 – Link Graph Analysis

6592 entries are associated with Link Graph Item 4. Notice how it seems that 123.123.100.165 is being attacked by over 4000 + packets on port 80. It is possible that these packets can be false positives but the presence of WatchList traffic makes this address suspect. If this is a webserver, then I would look to patch the Operating System and WebServer software.

	# off
SourceIP	**Occurances**
141.157.254.236	2238
141.157.253.155	1048
66.77.73.236	293
218.43.21.223	102
210.83.197.70	82
66.77.73.144	62
172.160.191.30	54
*	*
*	*
*	*
4000+	entries
63.93.99.28	1

Figure 12 – Link Graph Item 4 Occurrences

IV – Fighting Back

A – Affected Network Devices

The following are a list of IP addresses that should be investigated for evidence of unauthorized activity and/or services. Figure 6 shows questionable devices on the enterprise network based on how many unique outbound connections generated alerts.

	Unique
Dest IP address	**Sources**
123.123.100.165	2143
123.123.24.34	188
123.123.24.44	103
123.123.30.4	103
123.123.247.94	93
123.123.220.42	82
123.123.233.222	78

123.123.29.11	70
123.123.246.178	64
123.123.223.78	62
123.123.239.126	59
123.123.205.226	59

Figure 13 – Destination IP Address Connecting to Unique Sources

Based on the OOS file – the following is a list of internal IP addresses where they were the source address sending OOS packets out of the enterprise. These machines need to be inspected very carefully for intrusions.

Source IP address	# of Occurences
123.123.12.4	56
123.123.12.2	7
123.123.244.58	2
123.123.253.2	2
123.123.252.14	1

Figure 14 – OOS internal sourced packets

Based on the scan files the following are the source addresses associated with various scan types.

B – Defensive Recommendations

Install a stateful firewall. This device will cut down on basic probes, pings, and scans from the outside world.

Capture binary packet data and look for TCP SRC and DST packets. Note the MAC address and look for the sources of those addresses. These machines will be DDoS clients and will need to be carefully quarantined and cleaned.

At the perimeter router, block RFC 1918 addresses and unnecessary or dangerous protocols like RPC/TCP 111 or finger TCP 79.

Patch and update all OS - start with the servers and work down to the workstations. Default installs of operating systems are ripe with exploitable vulnerabilities.

Install Anti-Virus and keep the signatures updated. This will help in combating Trojan and Worm software.

For SubSeven Trojan machines:

http://www.whitehats.com/ids/trojan/

According to whitehats.com -

By default the Trojan uses TCP port 27374, but this is configurable by the configuration program.
It is normally distributed as a Win32 PE exe dropper that may be disguised as a JPG or BMP picture. When run, this dropper installs two files into the WINDOWS folder of the user's hard disk. These two files are the main server exe files, normally called "MSREXE.EXE", and a loader program normally called "RUN.EXE", "WINDOS.EXE" or "MUEEXE.EXE".

V – Methodology

A – Snortsnarf and MS Excel

I began using snortsnarf Perl script to perform a cursory analysis of the small data files. The large data files used too much memory and sytem resources so even my 2 GHZ/512MB RAM laptop died while trying to analyze the 155MB and 75MB files. This analysis revealed that there were a high level of portscan data present. Snortsnarf was only helpful in analyzing alert file data. MS Excell was also used to order and sort data, as well as create tables and graphs.

B – Custom script

I used a custom script to import the alert, scan, and OOS data into an MSSQL database. In order to do this properly, data delimitation had to be decided upon and tested. After several iterations, I was able to upload 155MB of scan data in 50 minutes.

C – MSSQL

Once the data was loaded into MSSQL, I ran over 250 queries testing and massaging the data. I was able to generate output that highlight the grandest of alerts as well as the most subtle. Here are 3 examples of queries I used and the data they output.

The simplest query would deliver the top occurrences of any row item, in this case, it is the source IP address – like that exhibited in Figure 4.

```
SELECT SourceIP, count (*) FROM alerts Group By SourceIP Order By 2 desc
```

This is a very simple line that produced very astounding results. This query took 3 minutes to run through 1.5 million rows of data,

The second query is a two step process. I was very interested in displaying how promiscuous a certain source IP address was in the enterprise network. I first had to compute how many connections existed from one source IP address to any other IP address, load that into another table sorted and grouped by the source IP address. The output looks like:

Source IP address	Dest IP address	Count IP
169.154.207.39	209.126.247.144	2
169.154.246.242	209.126.247.144	2
169.154.55.11	209.126.247.144	2
169.154.68.13	209.126.247.144	2
169.154.68.31	209.126.247.144	2

Figure 15 – Step 1 in Creating Unique Occurances Table

The code that generated this table looks like:

```
SELECT       SourceIP, DestIP, CountIP=count(*)
into     alerts1
from        alerts
            group by SourceIP, DestIP
            order by SourceIP , DestIP
```

The second step was to count every IP address that the Source address connected with along with counting the aggregate number of each occurrences (CountIP column).

The code for step 2 looks like:

```
SELECT       SourceIP, 'Total Connections'=sum(CountIP),'Unique Target
Addresses'=Count(*)

from        alerts1
            group by SourceIP
            order by count(*) desc
```

The last SQL script that proved invaluable allowed me to search and identify if an IP address had any conversations on ports listed in my top ten talkers:

```
SELECT * FROM scan where sourceip='123.123.87.44' and destport !='27005' and
destport !='43625'and destport !='80' and ( destport = '1214' or destport='65535' or
destport='32771' or destport='0'or destport='55850' or destport='1210' or
destport='137' or destport='80' or destport='1310' or destport is NULL)
```

D – Microsoft Visio

I used this program along with various SQL scripts to isolate and identify all talkers to and from a selected IP address.

VI Sources

http://www.arin.net/index.html

http://staff.washington.edu/dittrich/talks/core02/tools/tcpdump-filters.txt

http://www.cs.columbia.edu/ids/scam/

http://www.giac.org/gcia_study_guide_v33.pdf

http://www.thinkbrown.com/programming/sql_tutorial.pdf

http://www.wittys.com/files/all-ip-numbers.txt

http://www.whitehats.com/cgi/forum/messages.cgi

http://www.simovits.com/sve/nyhetsarkiv/1999/nyheter9902.html

http://www.shmoo.com/mail/ids/

http://www.cs.columbia.edu/jam/

http://www.unixreview.com/documents/s=1233/urm0107f/0107f.htm

http://ise.gmu.edu/~snoel/index_files/slide0001.htm

http://ise.gmu.edu/~snoel/index_files/slide0001.htm

http://www.silicondefense.com/support/snortsupport/

http://www.sans.org/resources/idfaq/oddports.php

http://www.sans.org/y2k/ports.htm

http://forum.sans.org/discus/messages/147/4532.html?1033753030

http://www.cs.columbia.edu/ids/

http://www.cs.columbia.edu/ids/dude/

http://www.freesoft.org/CIE/Course/index.htm

http://www.ntbugtraq.com/default.asp?pid=36&sid=1&A2=ind0005&L=ntbugtraq&F=&S=&P=10991

http://www.neohapsis.com/articles/default.php

http://www.ll.mit.edu/IST/ideval/index.html

http://www.cs.columbia.edu/ids/publications/

http://www.sans.org/rr/intrusion/logfile.php