Intrusion Detection: In depth

GIAC Certified Intrusion Analyst (GCIA)
Practical Assignment
Version 3.3

David M Lewis

8th November 2003

Table of Content:

Part 1 Describe the state of Intrusion Detection:- The need for Consolidation of Analysis Tools within a Multi Vendor Environment………………… 4
Introduction…………………………………………………………………4-1
The Consolidation/Correlation Process…………………………………..5-2
Installation……………………………………………………………….. 5-3
Normalization……………………………………………………………… 5-4
Correlation…………………………………………………………………... 6-5
Figure 1……………………………………………………………………. 7-5
Reporting……………………………………………………………………8-6
Conclusions…………………………………………………………………9-7
References………………………………………………………………..10-8
Part 2 Network Detects, Detect 1: Backdoor “Q”……………………………………11
Source of Trace…………………………………………………………….. 11-1
Figure 1…………………………………………………………………….. 11-1
Detect Was Generated By………………………………………………… 11-2
Snort rule that generated alert…………………………………………… 12-2
Alert generated by Eagle X IDS Centre………………………………… 12-2
Extracted from Snort 2.0 Alert File……………………………………… 13-2
Probability the Source address was spoofed……………………………… 13-3
Description of the Attack…………………………………………………… 14-4
Attack Mechanism…………………………………………………………… 15-5
Correlations…………………………………………………………………… 16-6
Evidence of Active Targeting………………………………………….. ……… 17-7
Severity……………………………………………………………………… 17-8
Defensive Recommendation……………………………………….. ……… 17-9
Test Question…………………………………………………………. …….. 18-10
Part 2 Network Detects, Detect 2: Land Attack (Land.c)…………………… ……… 18
Source of Trace…………………………………………………………………18-1
Figure 1…………………………………………………………………….. 18-2
Detect Was Generated By………………………………………………… 19-2
Snort rule that generated alert…………………………………………… 19-2
Alert generated by Eagle X IDS Centre………………………………… 19-2
Figure 2…………………………………………………………… 20-2
Extracted from Snort 2.0 Alert File……………………………………… 20-2
Probability the Source address was spoofed……………………………… 21-3
Description of the Attack…………………………………………………… 21-4
Figure 3…………………………………………………………………….. 22-4
Attack Mechanism…………………………………………………………… 22-5
Figure 4…………………………………………………………………….. 23-5
Correlations…………………………………………………………………… 24-6
Evidence of Active Targeting………………………………………….. ………24-7
Severity……………………………………………………………………………24-8
Defensive Recommendation……………………………………….. ………25-9
Test Question…………………………………………………………. ……..25-10
Part 2 Network Detects, Detect 3: RingZero……………………………….. ……… 25
Source of Trace…………………………………………………………….. 25-1
Figure 1………………………………………………………………… 26-1
Detect Was Generated By………………………………………………… 26-2
Snort rule that generated alert…………………………………………… 26-2

Alert generated by Eagle X IDS Centre……… 27-2
Extracted from Snort 2.0 Alert File……… 27-2
Extracted from Snort Port Scan Log……… 29-2
Probability the Source address was spoofed……… 30-3
Description of the Attack……… 30-4
Figure 2……… 31-4
Attack Mechanism……… 31-5
Correlations……… 32-6
Evidence of Active Targeting……… 32-7
Severity……… 32-8
Defensive Recommendation……… 33-9
Test Question……… 34-10
Part 3 The Scenario “This is life Jim, but not as we know it!”……… 34
The Data……… 34-1
Table 1……… 34-1
Executive Summary……… 34-2
Alert Signatures and Priorities……… 35-3
Table 2……… 35-3
Table 3……… 37-3
Table 4……… 37-3
Scan Signatures and Prioritization……… 38-4
Table 5……… 38-4
Table 6……… 38-4
OOS Signatures and Prioritization……… 39-5
Table 7……… 39-5
Table 8……… 39-5
Table 9……… 40-5
Alert Analysis……… 40
SMB Name Wildcard……… 40-6
Table 10……… 41-6
Chart 1……… 41-6
Table 11……… 42-6
Possible Trojan Server Activity……… 43-7
Table 12……… 43-7
Figure 1……… 44-7
TCP SRC and DST outside network……… 44-8
External RPC Call……… 47-9
SunRPC highport Access……… 47-10
Scan Analysis……… 48
SYN Scan Externally Based……… 48-11
Table 13……… 48-11
UDP Scan Externally Based……… 49-12
OOS Analysis……… 50
Scan Internally Based……… 50-13
Scan Internally Based……… 50-13
Scan Internally Based……… 51-15
Defensive Recommendations……… 51-16
References……… 52-17
Annex A: Tools Used to Complete this Paper……… 53

Part 1 - Describe the state of intrusion detection:- The need for Consolidation of analysis tools within a multi vendor environment.

1. Introduction.

The IT community has long seen the need for centralization of information within an enterprise level network; we have systems that monitor the serviceability of our network, we have centralized management systems to assist with maintenance and updates. We even centralize the fault reporting system to one helpdesk, this gives us a full picture of the status of our networks at any given time, all of which aims to prevent costly downtime.

So why not apply the same philosophy to our enterprise level management of our Intuition Detection Systems? In doing so, we are immediately faced with the challenge of the sheer volume of information being generated. We have to consider the source of this information, whether it has originated from the Intrusion Detection system (IDS), the Firewall, the Anti Virus (AV) or the Operating System logs, while analyzing the threat to our network. The time that it takes the Security Analyst to assess this risk/threat, and formulate a conclusion is a valuable commodity. This challenge becomes increasingly complex when we have a multi vendor environment with more than one flavour of IDS, Firewall, AV and OS, as the analyst has to interpret the differing responses to the same stimulus generated by a possible attack.

The need for an automated facility to assist the analyst with such an overwhelming challenge has increased along with the number of vulnerabilities/attacks that are discovered on a daily basis. We, as analysts have to find a way of bringing all the generated alerts together, interpreting them, and then presenting them in a logical and understandable manner. This process will benefit greatly from automation, and a number of vendors have provided solutions to assist in the consolidation of this vast amount of information into the one arena.

There are a large number of vendors producing consolidation tools in the market place to date, but they have all originated from the same necessities; the need for centralized, correlated information. The aim of these vendors has been to present the analyst with a normalized reporting system, enabling the analyst to snapshot the full enterprise security picture at any given time, to reduce the amount of false positives generated, and to increasing the speed and accuracy of the analysis while assessing any possible attack.

The Garner Group has summarised this type of event consolidation into the “5 C’s”, which can provide a good rule of thumb while considering these types of tools.

a. Collection
b. Consolidation
c. Correlation
d. Control

e. Communication

2. The Consolidation/Correlation process

It was the pioneers of the IT security field that first discovered the need for consolidation of information and the first tool that was ever used for the normalization of the information presented by the IDS, Firewall, AV and OS logs was the humble and lowly Security Analyst his/her self. It was he/she that collated the information generated by the above devices; it was he/she who drew the first graphical representation of possible attack trends. It was he/she that processed this information, collating it into groups of attack types, and cross referred with IP information to assess the source, target and threat. It was also he/she that realized the need for automation, and soon put the systems in place to produce reports/graphs to assist them in their work. From its humble beginnings, the consolidation tool has developed from a concept to a valid and important product.

3. Installation:

This phase of the deployment is the most technical, time consuming and frustrating of all. Both the vendor installation team and the IDS management team have to prepare considerable documentation to provide a complete overview of the network topology. You will have to consider the following topics while gathering the necessary information for your documentation, as all of this information will be used to configure both the collection and normalization engine.

a. Number of subnets, (in a Basic class A, this could be up to 254 subnets, 65534 host per subnet).
b. Types/configuration of IDS system and Vendor
c. Types/configuration of Firewall and Vendor
d. Type/configuration of AV and Vendor
e. Types/configuration of OS/System logs used
f. The type of data that will be emitted from the each device, i.e. SNMP, SMTP or Syslog
g. Have the OS's been hardened?

4. Normalization

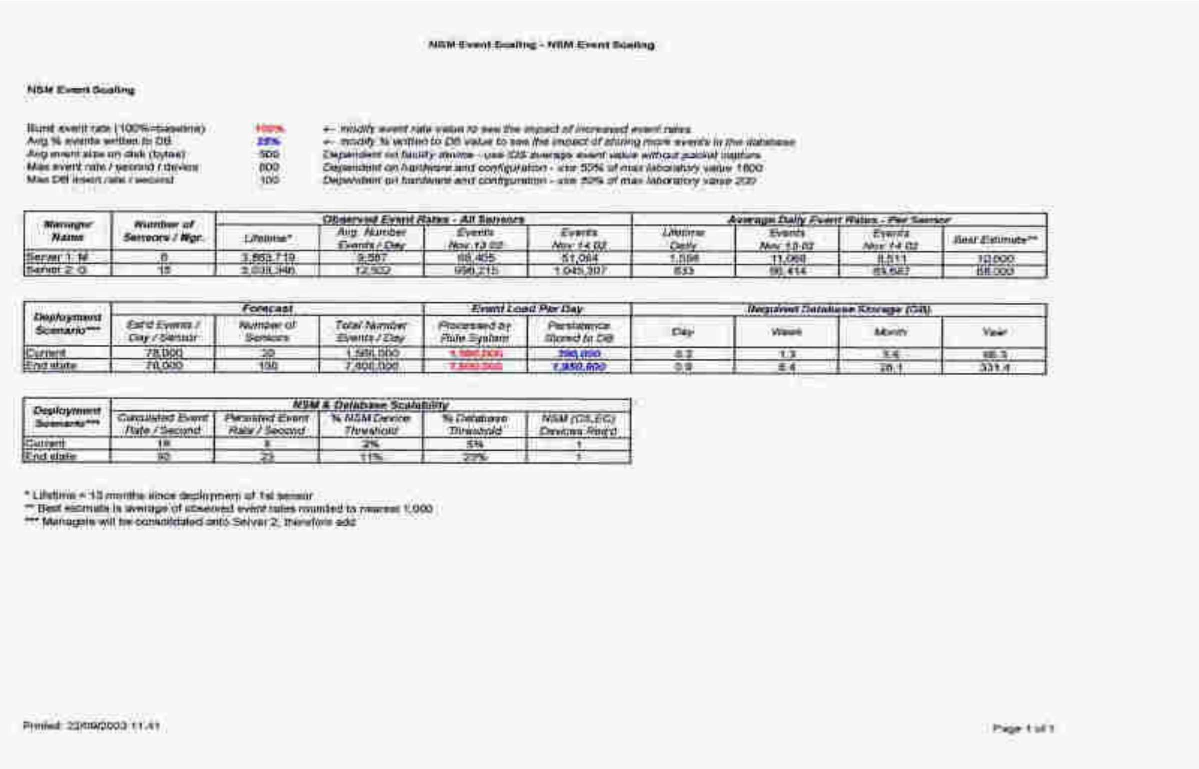
This is the first phase of the consolidation process, and is the means by which the events from the multiple vendors are all translated into a common format. With the development of XML, the consolidation industry have adopted it as the translation language of choice, and it is used to pipe all the events generated by the multiple IDS's, Firewalls, AV and System logs into their engines/systems before processing takes place. (I will not cover XML in detail, as this has already been covered by Michael Dawson in his GIAC paper, Infopeople Security Solutions, Dec 7 2002, in which he describes the use of XML in conjunction with the consolidation tool Intellitactics). As the normalization is completed by the event collectors, you can further mitigate the load, and balance the input through the network. Some vendors advocate pre-processing to provide a distributed solution and thus reduce the bandwidth. Others prefer to process the information in memory, this again produces overheads on the systems used (You will need as much RAM as possible in the system), but the vendors advise that this will greatly enhance the real time event reporting. You will have to assess which options will benefit your network, as this could produce long team issues on the reporting and monitoring system.

During the normalization phase each event is given a generic alert, these alerts contain the following information:

a. Source address
b. Destination address
c. Event severity
d. Event ID
e. Event category
f. Timestamp
g. Protocols.

5. Correlation

Once the events have passed thought the normalization phase, it is the job of the event collector to process the vast amount of data, and apply both rule based and protocol based analysis to the data. The rules based system used computational model filtering to assess the event against known forms of attack pattern, although this can be an accurate way of processing the event. It does, however, have a large impact on resources of the machine, as all traffic passing the sensor has to be compared to the known filters, this information also has to be stored. The storage issue may not be of great concern with the falling cost of hardware storage, but in a large enterprise environment you could end up with a data silo, as opposed to a storage system. To give some indication of the amount of storage necessary, Intellitactics have provided a spreadsheet tool to assist in the calculations, and example of its output is at figure 1.



NSM Event Scaling - NSM Event Scaling

NSM Event Scaling

Burst event rate (100%=baseline)	100%	← modify event rate value to see the impact of increased event rates
Avg % events written to DB	25%	← modify % written to DB value to see the impact of storing more events in the database
Avg event size on disk (bytes)	500	Dependent on facility device - use IDS average event value without packet capture
Max event rate / second / device	800	Dependent on hardware and configuration - use 50% of max laboratory value 1600
Max DB insert rate / second	100	Dependent on hardware and configuration - use 50% of max laboratory value 200

Manager Name	Number of Sensors / Mgr.	Observed Event Rates - All Sensors				Average Daily Event Rates - Per Sensor			
		Lifetime*	Avg. Number Events / Day	Events Nov 13 02	Events Nov 14 02	Lifetime Daily	Events Nov 13 02	Events Nov 14 02	Best Estimate**
Server 1: M	6	3,863,719	9,587	68,405	51,084	1,598	11,068	8,511	10,000
Server 2: O	15	3,038,346	12,502	998,215	1,045,307	833	66,414	69,687	68,000

Deployment Scenario***	Forecast			Event Load Per Day		Required Database Storage (GB)			
	Est'd Events / Day / Sensor	Number of Sensors	Total Number Events / Day	Processed by Rule System	Persistence Stored to DB	Day	Week	Month	Year
Current	78,000	20	1,560,000	1,560,000	390,000	0.2	1.3	5.6	68.3
End state	78,000	100	7,800,000	7,800,000	1,950,000	0.9	6.4	28.1	331.4

Deployment Scenario***	NSM & Database Scalability				
	Calculated Event Rate / Second	Persisted Event Rate / Second	% NSM Device Threshold	% Database Threshold	NSM (CS,EC) Devices Req'd
Current	18	5	2%	5%	1
End state	90	23	11%	23%	1

* Lifetime = 13 months since deployment of 1st sensor
** Best estimate is average of observed event rates rounded to nearest 1,000
*** Managers will be consolidated onto Server 2, therefore add

Printed: 22/09/2003 11:41

Page 1 of 1

Figure 1.

A scoring process similar to that used within the IDS Signatures and Analysis, Part 1 Chapter 4, is also used, where the severity of an event is assessed by calculating how relevant the attack may be. During this calculation, consideration is given as to whether the attack is relevant to the OS being used, the topology of the network and criticality of the system being targeted. This will give a numerical reference as to the overall severity of the event, i.e. If the target was a Windows box, and the attack was a Windows based attack, with little or no security in place, then the numerical number would be high, ((Critical + Lethal) – Countermeasures = Severity, therefore (3+4) – (1+2) = 4, which illustrates a high risk to this specific system).

The correlation phase, also takes into consideration the different types of report from the Security system within the network. If we look at the many vendors who build IDS systems today, and the different way in which each reports a similar event, we can quickly see that even the experienced analyst will be overwhelmed by the amount of different data produced. Most of the IDS vendors use similar, but not standardized, reporting for many events, i.e. malicious software, buffer overflow and unauthorized access. The correlation engines used within the different types of tools available, aims to associate the multiple event types and the multiple sources, across multiple network nodes, regardless of origin. This will provide:

a. Event sequencing by comparing short and long term events

b. Event persistence by assessing network loading

c. Event Directed data collection

Thus easing the workload of the analyst. This process reduces the amount of data from an estimated 20,000 -30,000 events, to a possible 100 event alarm ID's/categories. By collating these alarm types into one basket, it will produce a data, or graphical representation of the network activity. As the analyst will not have to spend his precious time, calculating and grouping the event type manually. This greatly reduces the amount of false positives generated, while the identification and escalation of the stealthy, blended attacks will reduce the false negatives and thus increase the speed and accuracy of the data analysis/reporting.

6. Reporting

At this point in the process, we need to have some output for all of the data input, and that comes from the reporting console. The reporting console is the analysts interface to the reporting system, and will allow him/her to displaying the input in a human friendly format; using either a text based or graphical representation in real time, or as near to real time as the possessing power dictates. The speed of the interface between correlation engine and reporting console is an important component within the system, thus the faster the interface, the smaller the response time to an incident. By reducing the response time, by early detection of trends and attacks, we reduce both the financial costs to our network and our business, and the physical cost of repair and recovery. These costs will rise due to the following:

a. Rebuilding systems

b. Implementing new security policies

c. Changing User passwords

d. Applying Patches

e. Loss of historical logs

f. Reload backups

g. Loss of customer confidence

If we consider the recent outbreak of the MSNachie worm, and the speed of propagation, it is evident that some IDS's reported the infection of the system (TFTP data transfers) before identifying the RPC buffer overflow vulnerability to the system (possibly due to thresholding). Using the real time reporting console an analyst could produce an event report of all TFTP data transfers, and cross refer to the later RPC buffer overflows, and quickly identify possible infected or attempted infection of system within the network. However, we can add the firewall and AV information to this event, by identifying any port 135 firewall drops, and host AV reporting infections/detections of the above worm. All this information can be used to track the propagation of such a worm through your network.

As the data is stored within a centralized database, it is possible for the analyst to produce a vast array of report types. This fully customisable feature is a strong selling point for management; as it can quickly produce an overview of the network activities, which can be used for presentational purposes, and in most cases can be fully tailored to the audience's knowledge and understanding of network security.

This database can also be utilised for further analysis of the network traffic which We can either consider as, trend analysis or advances analysis, as we attempt to gain more information of the types of traffic being produced on our network. If we consider advanced analysis, or the search for the proverbial needle in the haystack, we may need more information than is contained within such a system (i.e. packet analysis), however, this tool can provide an in-depth overview of the network activities. During advanced analysis, we can identify a trends or regular events and either dismiss them as false positives, thus tuning the sensors or the correlation tool (sometimes both) accordingly. The flip side will be the identification of a malicious activity against our network, or even, our network against competitors/individuals systems, either of which can be a potential compromise/embarrassment of/to our system, or will need further investigation. This type of analysis can be undertaken retrospectively, or in conjunction with a live event.

7. Conclusions

When first presented with the concept of consolidation tools, I was extremely excited. I could see the need for such a tool in the arsenal of the IDS Analyst. The fact that all the information would be automatically displayed in a concise and user friendly manner greatly impressed me. Having now worked extensively with such products, I have found myself going though the following thought processes, from excitement, to apprehension, to overconfidence and now understanding.

It is a common misconception, mainly by the management that the training needs of the new analyst can be reduced with the availability/installation of such tools, this may be due to the graphical displays, the presentation of information and the sales pitch from the vendors. This conception quickly dissolves during the installation and configuration phase, and the realization that more training maybe necessary to further understand the new systems as well as the old.

There is still one factor that has to be considered, and that is, no matter what tools we place on our desk tops, we can never replace the human factor. The experienced security analyst has a great advantage over any automated tool, and that is the gut instinct. The ability to see beyond the glossy front end, this factor can not be full explained, but could be considered aptitude. This aptitude is the ability to feel and understand the many moods of his/her network, and listen when it is happy, and be caring and understanding when it is not, and above all else, to protect it while it is being violated or attacked.

8. References:

Intellitactics:- Panning and Preparation Manual:-
http://www.intellitactics.com/index.cfm
NetForensics:-
SANS IDS Signatures and Analysis, Part 1
eCommSecurity :-
http://www.ecommsecurity.com/apollo.html
LAN Logic, Symantec ManHunt:-
http://www.lanlogic.com/security/intrusion-detection.asp
Security Profiling:-
http://www.securityprofiling.com/
CIO Information Network, Security Threat Correlation: The Next Battlefield :-
http://www.cioupdate.com/reports/article.php/1501001
SourceFire, Sourcefire unleashes indursrtry's most comprehensive intrusion management system:-
http://www.sourcefire.com/pressoffice/press_releases/pr021209.htm
ASP News, The Need for Real-Time Threat Correlation:-
http://www.aspnews.com/strategies/article/0,2350,9921_1545771,00.html
Tenable, Real-Time Intrusion Detection and Vulnerability Correlation:-
http://www.tenablesecurity.com/ids-va.html
USJFCOM, Network Security Management Correlation and Display (NSMC&D):-
http://www.jfcom.mil/about/experiments/mc02/concepts/nsmc.htm
Applied Watch Command Centre, Enterprise Command and Control for the Snort IDS:-
http://www.appliedwatch.com/
Information Security, eTrust Security Command Center:-
http://infosecuritymag.techtarget.com/ss/0,295796,sid6_iss21_art102,00.html
Network Computing, Security Information Management Tools: NetForensics Leads a Weary Fleet:-
http://www.networkcomputing.com/1307/1307f22.html
Security Focus, IDS Correlation of VA Data and IDS Alerts:-
http://www.securityfocus.com/infocus/1708
CNN, Taking Event Correlation Seriously, by Dennis Dorgseth:-
http://www.cnn.com/2000/TECH/computing/03/20/event.corr.idg/
E-Security:-
http://www.esecurityinc.com/products/correlation.asp
Secure Commerce Systems, GuardTower Security Correlation Appliance
http://www.securecommercesystems.com/guardtower.html
FCW.COM, Security Overload by Rutrell Yasin:-
http://www.fcw.com/fcw/articles/2002/0812/cov-sec-08-12-02.asp
Computer World, Sidebar: Security Event Management Tools By Jaikumar Vijayan:-
http://www.computerworld.com/securitytopics/security/story/0,10801,84105,00.html

Assignment #2: Network Detects
Detect 1: Backdoor "Q"

1. Source of Trace.

The raw packet capture used for this exercise was obtained from http://www.incidents.org/logs/Raw, and was selected from the 2002.5.10 capture. From the analysis of the data, I consider the network layout to be as detailed at figure 1. This conclusion is made as the source address and destination IP addresses are associated with the MAC address which falls within the range used by CISCO, and under standard network configuration the IP source will take on the MAC address of the last router the packet is passed through.

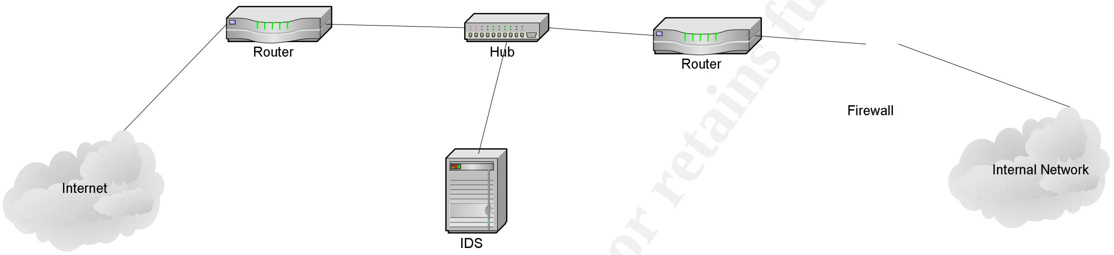


Figure 1

2. Detect was generated by:

The detection suite that I used for this exercise was as follows:

a. A default installation of the Snort 2.0 build sensor, with standard logging, alert file generation and rule sets. http://www.snort.org/dl/binaries/win32

b. Windump, http://windump.polito.it/install/default.htm

c. Ethereal 0.9.11, http://www.ethereal.com/download.html

d. Eagle X IDSCentre 1.1 RC4 Installation, http://www.engagesecurity.com/downloads

   (Note: The standard fit for this build is a Snort 2.1 engine, I changed this to a 2.0 engine due to a possible bug in the later version while replaying events (-r). The snort 2.1 did not generate the expected alerts when collated with the output of Windump and Ethereal).

The reasons for choosing this suite of tools, boils down to familiarity. The Snort 2.0 build was chosen as a base line sensor with which to test the output from the Eagle X build. The Eagle X IDS Centre was chosen to provide a user friendly tool for correlation of events, and to make use of the PHP facilities and potted queries, thus reducing the workload necessary on the volume of information generated. Both

Windump and Ethereal were used to analyse the individual packet content, cross referring information to confirm the validity of my finding, and identifying any false positives.

Snort rule that Generated alert:

alert tcp 255.255.255.0/24 any -> $HOME_NET any (msg:"BACKDOOR Q access"; flags:A+; dsize: >1; reference:arachnids,203; sid:184; classtype:misc-activity; rev:3;)

Alert Generated by Eagle X IDSCentre

Extracted from the Unique alert list.

Signature	Class	# EVT	#Unique	Start time	End time
BACKDOOR Q access	misc-activity	43 (12%)	43	2002-06-10 01:18:48	2002-06-11 00:52:36

Extracted from Query Results by drilling down through "Traffic Profile by Protocol"

Signature	Time	Source IP	Dest IP	Prot
BACKDOOR Q access	2002-06-10 15:10:00	255.255.255.255:31337	46.5.89.229:515	TCP

Extracted from a Query Result on above ID # for event (not shown)

Meta	ID #	Time	Triggered Signature
	2 - 203	2002-06-10 15:10:00	[arachNIDS][snort] BACKDOOR Q access

Meta	Sensor	name	interface	filter
		[reading from a file]	[reading from a file]	*none*
	Alert Group	*none*		

IP	source addr	dest addr	Ver	Hdr Len	TOS	length	ID	flags	offset	TTL	chksum
	255.255.255.255	46.5.89.229	4	5	0	43	0	0	0	14	11497

IP	FQDN	Source Name	Dest. Name
			Unable to resolve address
	Options	*none*	

TCP	source	dest	R	R	U	A	P	R	S	F	seq #	ack	offset	res	window	urp	chksum

port	port	1	0	R G	C K	S H	S T	Y N	I N							
31337	515				X		X			0	0	5	0	0	0	57616

Options	*none*

Payload

```
 length = 3

000 : 63 6B 6F                                                  cko
```

The above output is an HTML, user friendly, representation of the Hexadecimal/ASCII dump of the Snort capture.

Extracted from Snort 2.0 (Default Installation) Alert file;

```
[**] [1:184:3] BACKDOOR Q access [**]
[Classification: Misc activity] [Priority: 3]
06/10-01:18:48.944488 0:3:E3:D9:26:C0 -> 0:0:C:4:B2:33 type:0x800 len:0x3C
255.255.255.255:31337 -> 46.5.87.61:515 TCP TTL:14 TOS:0x0 ID:0 IpLen:20
DgmLen:43
***A*R** Seq: 0x0  Ack: 0x0  Win: 0x0  TcpLen: 20
[Xref => http://www.whitehats.com/info/IDS203]
```

3. Probability the Source address was spoofed

The probability that this packet contains a spoofed source address is High, and blatant. I have considered that this packet has been generated from outside the network, as the source MAC is associated with the allocation which falls within the range used by CISCO, (we could consider that this too could be spoofed, but I feel that this is not the case) therefore the packet has originated from the internet facing gateway. The use of the broadcast address within a packet generated from outside the perimeter is against standard practices used within an Ethernet/Routed network; as such traffic will be confined within the originating network, as the Router will not pass traffic address to the broadcast address outside the subnet.

Other factors can be considered within this packet that would identify it as being crafted.

a. The use of the Ack flag in conjunction with the Reset flag is not standard practice.

b. A low TTL, if this was a legal packet; I would expect it to have a higher TTL. This assumption comes from the fact that most O/S's have published TTL's, and by applying that knowledge here and comparing this with your network topology you should expect a much higher figure than 14. i.e. A standard windows TTL is 128, to achieve a TTL of 14, this packet would have had to traversed 114 routers, does your network have this many routers?

c. The Sequence Numbers and Acknowledgment number are all set to 0.

d. Both the IP header and the TCP checksums are incorrect (identified using Ethereal)

e. The constant use of port 31337 (eleet), which is considered to be a well known port used by Trojans, and will stimulate the IDS to report activity associated with this port.

4. Description of the Attack

Research had indicated that "Q" is a Tool originally Written by Mixter, which is described by Packet Storm Security http://packetstormsecurity.nl/groups/mixter/ as:

> ”Q v2.4 is a client / server backdoor which features remote shell access with strong encryption for root and normal users, and a encrypted on-demand tcp relay/bouncer that supports encrypted sessions with normal clients using the included tunnelling daemon. Also has stealth features like activation via raw packets, syslog spoofing, and single on-demand sessions with variable ports. Changes: Now uses strong RSA/libiSSL encryption for sessions; compatibility with libmix1.2; many bugfixes. Homepage: http://mixter.void.ru. By Mixter”

There has been much speculation about this event, both within the GIAC certification program, and the security community as a whole. This speculation has spanned from this event being a failed worm to "completely harmless" as quoted by Crist J Clark, Network security Engineer, Globalstar.

The attack is based around the above tool, which is legitimately used as a remote access administration tool, in the same vein as Net Cat and VNC. The tool is designed to give the network administrator remote root access on a Unix based operating system. However, if the client is identified and compromised, this will give an attacker the same privileges, with dire consequences.

The specific use of this tool is aimed at identifying a client that has been compromised, or by identifying a Trojan'd version of the client. The second part of this statement I consider to be more relevant within this context, this is due to the structure of the attack.

The aim of this crafted packet is to obtain access to the network, by deceiving the firewall. It is hoped by the attacker, that the firewall will ignore the 255.255.255.255 address and assume that it originated from within the network. It also targets a known port (515) which is used for the spooler services, this is an attempt to again compromise the firewall rule set, as traffic to this port may be considered to be legitimate printer traffic by the firewall administrator, and be allowed through. We also have to consider the use of the Ack and Rst flags; this would suggest that the attacker is attempting to disguise the packet as a replay to a request from within the target network, thus attempting to circumvent any stateful firewall.

Other factors that identify the packet as being crafted, such as the TTL, Sequence #, Acknowledgement # and the checksum could be considered to be inert, and due to the use of a packet crafting tool such as Rafale.

The pattern of the attack is such that I would consider there to be a single source, due to the errors within the crafted packet remaining constant, but I would stress that this is an assumption. Using this assumption it would appear that the attacker is attempting to enumerate the network for any "Q" clients, but the use of the broadcast address precludes direct replies. Therefore the question becomes "why".

5. Attack Mechanism

It is my assessment that, a Trojan has been developed around the "Q" tool, and that this tool has been disseminated around the hacker community, mainly via IRC's as this is possibly an easy way of propagation, and then used for malicious purposes, such as DDOS. The vehicle for the transportation of this Trojan could just as easily be by e-mail or an HTML webpage, and targeted against un-patched/out of date versions of Internet Explorer 5 and Outlook/Outlook express, as older versions will run code arbitrarily without interaction with the user. The communication or trigger for an event to be started would be as simple as the code within the payload of these packets (cko), which could be a command coded into the Trojan to stimulate a response from a compromised machine to a predefined IP address. This will bypass the security of a stateful firewall, as the connection has been originated from within the LAN and would be considered legitimate traffic. If we now consider that this is not a tool being used by a "Script Kiddie", and instead, consider that this is a well configured piece of code, it would be my assessment that the crafting of these packets has been configured in this way for a purpose.

If we now consider the emerging use of encryption within our networks, and via the Internet, between hosts (peer to peer) or within a VPN, we would have to take this into account if we were to craft packets to control a remote Trojan. I believe that the originator is aware of such problems and has an understanding of the encryption and its structure. They would have considered the ramifications of falsely triggering the carefully distributed malicious network they have built and the increased possibility that this string could be generated within the randomness of an encrypted packet, thus triggering the event pre-mutually, therefore a safety would be applied. That safety could be the reason for the low TTL, i.e. the TTL >=20 , which may not be expected under normal circumstances, couple this with the constant use of port 31337 and the safety margin is increased. This type of safety has recently been utilised within the CISCO vulnerability (Cisco IOS Interface Blocked by IPv4 Packets), where the TTL had to be >2 for the exploit to trigger, thus allowing the attacker to specify accurately the target for the attack.

If this type of tool were to be used for contacting a Trojan system, and the aim was to steal or gather information from the target network, the same safety protocol could be used to prevent the attacker from being DOS'd themselves. The "cko" command, which could be the stimulus used to initiate the connection back to the attacker's machine could inadvertent trigger a large number of clients that would overwhelm the receiving host.

I could not find any evidence of compromised clients attempting to initiate an outgoing connection within this packet capture. I suspect that there is a firewall inside of the gateway router as illustrated in figure 1 and that this is blocking and dropping these events. Even if a compromised client were to reply, the network configuration would preclude the connection passing the router due to the use of the broadcast address. I would suggest that this is an activation packet for the Trojan and that the Trojan is programmed to automatically send packets back to the "master" as described previously, and does not require a reply. I would also consider whether there were any UNIX based OS's within the scanned range. If not, then the threat would be lower. Therefore I would advocate that the systems integrity check be undertaken to assess if the system files had been altered. This would identify the possibility of a Trojan on the suspected system.

6. Correlations:

Snort Signature reference
http://www.snort.org/snort-db/sid.html?sid=184

Whitehats – The Intrusion Event Database reference
http://www.whitehats.com/info/ids203

Security Focus Postings with reference to similar traffic
http://online.securityfocus.com/archive/75/182244/2002-11-4/2002-11-10/1

insecure.org mailing list
http://lists.insecure.org/lists/incidents/2001/Jun/0265.html

Cisco IOS Interface Blocked by IPv4 Packets
Document ID: 44020 Revision 1.14
http://www.cisco.com/warp/public/707/cisco-sa-blocked.shtml

GCIA paper with similar traffic
http://www.giac.org/practical/Trenton_Riddell_GCIA.doc
http://www.giac.org/practical/GCIA/Al_Maslowski-Yerges_GCIA.pdf

7. Evidence of Active Targeting

There is evidence to support the possibility that attacker is specifically targeting this network. This attack has used 43 unique and random IP addresses within this network range, which would suggest that the attacker is looking for active hosts within this range, lost numbers as opposed to wrong numbers. However, we may not be seeing the whole picture as we are unsure how the router on the external side of the firewall is filtering traffic due to the action of the router in front of our firewall.

8. Severity

The severity for this event would be based on the assessment that there is a Unix based OS within the network, which could be susceptible to this type of attack, with no system countermeasures in place, but there is a network firewall present. Therefore, the following values will reflect this.

Severity = (criticality+lethality) – (System countermeasures + network countermeasures)

Therefore

Severity = (2 + 4) – (2 + 4)
Severity = 6 – 6 = 0

This would indicate that the threat to the network is low, and the scan is more of an annoyance that a threat. However, vigilance is still recommended.

9. Defensive Recommendation

The recommendations that I would make to prevent this tool from exploiting a network would be as follows:

a. Install/maintain a stateful firewall, ensuring that only ports that are needed for interaction with the Internet are open. Ensure anti spoofing is activated on the firewall this will drop all internal addresses that are generated from the external network.

b. Check and maintain the patch level of the system, paying particular attention to critical machines.

c. Install/maintain a system integrity checker, i.e. Tripwire, to highlight any changes that are made without authorisation.

d. Install/Maintain/monitor NIDS (Network Intrusion Detection System)behind the firewall, consider HIDS (Host Intrusion Detection System) on critical systems. This will identify any compromise that may take place due to a zero day event or firewall failure.

e. Limit information leakage/enumeration for the network by Web Servers, DNS servers and limit NetBIOS (if used) to within the network segment.

f. Maintain system auditing, check and monitor such event logs.

g. Install/maintain an Antivirus solution.

10. Test Question:

The use of the source IP 255.255.255.255 has been highlighted as inconsistent with Normal network activity, Why?

a. It is a Broadcast Address.
b. It is will not be passed through the router, to the external/internal network.
c. Will be received by all workstation connected to the internet.
d. It will be blocked by the firewall.

Answer B, the router will not pass broadcast traffic outside its subnet. Some may consider that D may also be correct, but the configuration of the router will prevent the traffic ever getting to the Firewall, this eventuality must still be addressed within the firewall policy, just in case.

Assignment #3: Network Detects
Detect 2: Land Attack (Land.c)

1. Source of Trace.

The raw packet capture used for this exercise was obtained from http://www.incidents.org/logs/Raw, and was selected from the 2002.5.10 capture. From the analysis of the data, I consider the network layout to be as detailed at figure 1. This conclusion is made as the source address and destination IP addresses are associated with the MAC address which falls within the range used by CISCO, and under standard network configuration the IP source will take on the MAC address of the last router the packet is passed through.

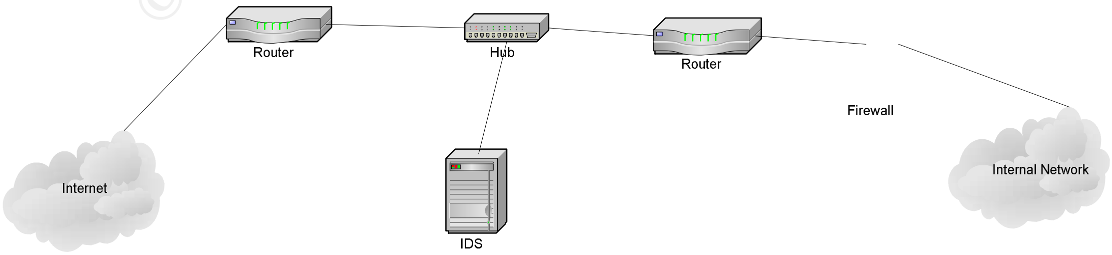


Figure 1

2. Detect was generated by:

The detection suite that I used for this exercise was as follows:

a. A default installation of the Snort 2.0 build sensor, with standard logging, alert file generation and rule sets. http://www.snort.org/dl/binaries/win32

b. Windump, http://windump.polito.it/install/default.htm

c. Ethereal 0.9.11, http://www.ethereal.com/download.html

d. Eagle X IDSCentre 1.1 RC4 Installation, http://www.engagesecurity.com/downloads

   The reasons for choosing this suite of tools, boils down to familiarity. The Snort 2.0 build was chosen as a base line sensor with which to test the output from the Eagle X build. The Eagle X IDS Centre was chosen to provide a user friendly tool for correlation of events, and to make use of the PHP facilities and potted queries, thus reducing the workload necessary on the volume of information generated. Both Windump and Ethereal were used to analyse the individual packet content, cross referring information to confirm the validity of my finding, and identifying any false positives.

Snort rule that Generated alert:

alert ip any any -> any any (msg:"BAD-TRAFFIC same SRC/DST"; sameip; reference:cve,CVE-1999-0016; reference:url,www.cert.org/advisories/CA-1997-28.html; classtype:bad-unknown; sid:527; rev:4;) alert ip any any -> any any (msg:"BAD-TRAFFIC same SRC/DST"; sameip; reference:cve,CVE-1999-0016; reference:url,www.cert.org/advisories/CA-1997-28.html; classtype:bad-unknown; sid:527; rev:4;)

Alert Generated by Eagle X IDSCentre

Extracted from the Unique alert list.

Signature	Class	# EVT	#Unique	Start time	End time
BAD-TRAFFIC same SRC/DST	bad-unknown	33 (94%)	33	2002-11-13 07:22:18	2002-11-13 23:25:13

Extracted from Query Results by drilling down through “Traffic Profile by Protocol”

Signature	Time	Source IP	Dest IP	Prot
BAD-TRAFFIC same SRC/DST	2002-11-13 07:22:18	207.166.38.167	207.166.38.167	IGMP

Extracted from a Query Result on above ID # for event (not shown)

Meta

ID #	Time	Triggered Signature
2 - 2	2002-11-13 07:22:18	url[cve][icat][snort] BAD-TRAFFIC same SRC/DST

Sensor	name	interface	filter
	DEFIANT:[reading from a file]	[reading from a file]	*none*

Alert Group	*none*

IP

source addr	dest addr	Ver	Hdr Len	TOS	length	ID	flags	offset	TTL	chksum
207.166.38.167	207.166.38.167	4	5	0	28	0	0	0	47	14296

FQDN	Source Name	Dest. Name
	Unable to resolve address	*Unable to resolve address*

Options	*none*

Payload

```
 length = 8

000 : 11 64 FB 08 F0 00 03 92                           .d......
```

Figure 2.

The above output is an HTML, user friendly, representation of the Hexadecimal/ASCII dump of the Snort capture.

Extracted from Snort 2.0 (Default Installation) Alert file;

[**] [1:527:4] BAD-TRAFFIC same SRC/DST [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
11/13-00:25:13.716507 207.166.97.158 -> 207.166.97.158
PROTO002 TTL:47 TOS:0x0 ID:0 IpLen:20 DgmLen:28
[Xref => http://www.cert.org/advisories/CA-1997-28.html][Xref => http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0016]

3. Probability the Source address was spoofed

The probability that this packet contains a spoofed source address is high. The packet contains the same source and destination address, and is being generated from outside the internet facing router. This is against standard practices used within an Ethernet/Routed network, as a packet will never be sent and received by the same IP address.

Other factors can be considered within this packet that would identify it as being crafted is the fact that the IP header checksum is incorrect, but the IGMP checksum is correct (identified using Ethereal). Due to this fact, I would assess that only the IP header has been spoofed and the IGMP header has been cut and pasted from a valid membership query.

4. Description of the Attack

This attack exploits vulnerability within the TCP/IP implementation. As stated within the paper published by CISCO (www.cisco.com/warp/public/770/land-pub.shtml) the attacker attempts to DOS the system by sending a single TCP SYN packet which contains the same source and destination IP address, while using the same target port as the source.

This single TCP SYN request will initiate a connection attempt within the system, and cause a loop failure due to the use of the same source and destination IP addresses. Within a windows 9x environment this will cause the system to crash, but will only slow down NT boxes. This attack is not limited to Windows system; versions of the CISCO IOS O/S, SunOS (precursor to Solaris), NetBSD and FreeBSD are also affected, either causing them to crash or slowdown.

The above implementation is a variation to the Land Attack. We still have the same source and destination address, but the protocol used is not TCP, but IGMP (Internet Group Management Protocol). This protocol is defined in RFC 1112, as the standard for IP Multicasting within the internet environment. This protocol allows a host to inform its local router, by the use of Host Membership Reports that it wants to be part of a specific group, and receive multicast messages addressed to that group. The host groups are identified by a class D IP address (1110 as the high order bits). Within the internet, the host group address ranges from 224.0.0.0 to 239.255.255.255, 224.0.0.0 is not used, and 224.0.0.1 is assigned to the “permanent group” of all IP hosts, including gateways (RFC 3330, IPv4 Addresses and RFC 3171 Refer).

In this attack, the IGMP message is a type 1 or a Host Membership Query, which is used to discover which host groups have members of specific groups within the attached local area networks. The RFC 1112 requires that the standard query is addressed to the all-hosts group (address 244.0.0.1.), with a TTL set to 1 (to remain within the LAN) or a minimum TTL for remote site connections. It also requires that the length of a request is 8 bytes, which will give a total packet size of 28 bytes (including the IP header). This will stimulate a membership report to be produced by the receiving hosts and returned to the originator.

When a multicast enabled router receives a membership query it will use one of two approaches, or modes to pass the information onto the next segment, or LAN. Their modes are sparse and dense. The dense mode is considered the best mode to use within a subnet'd environment, and will flood multicast datagram's into the subnet or LAN for maximum coverage. If the router has received an earlier membership query for the same group, it will remember which interfaces that the membership group resides on and will only flood that interface with the request. If it has not received a request for a known membership group it will flood all interfaces to obtain a response (figure 3)

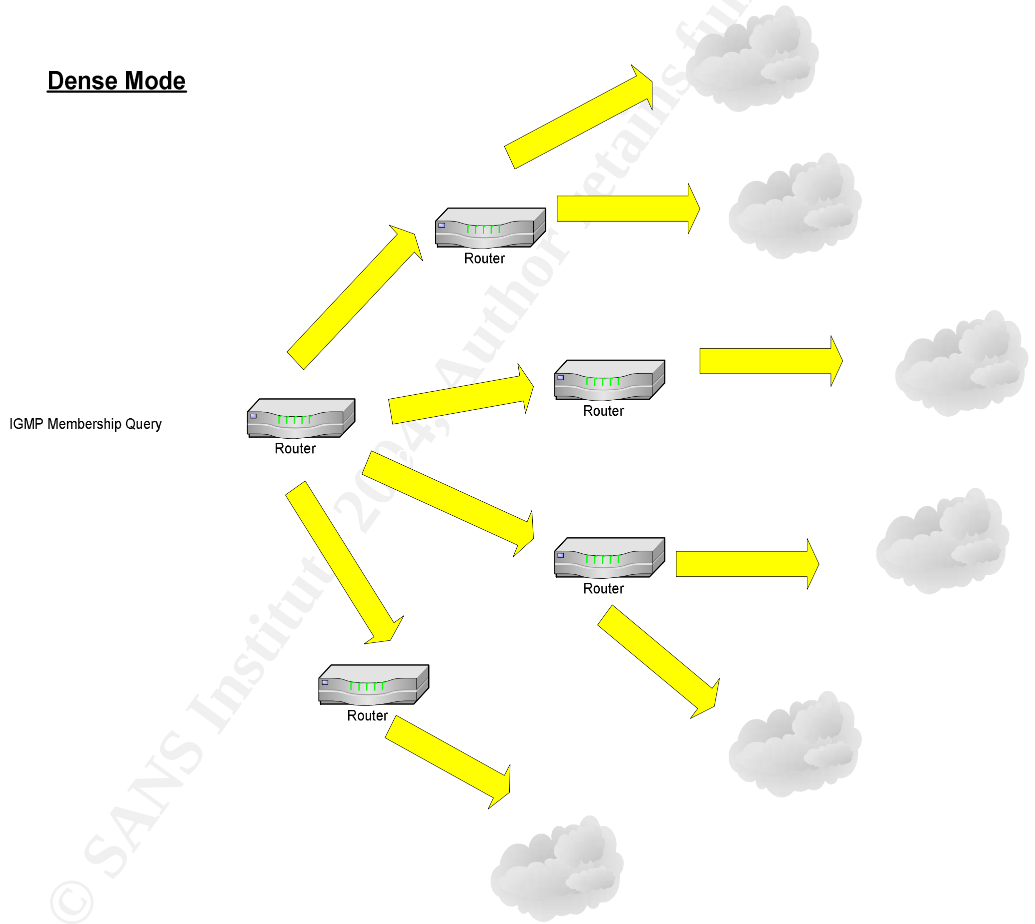


Figure 3

5. Attack Mechanism

The packet use for this attack has been crafted with the destination and source address of 207.166.38.167, and a TTL of 47. as it is a type 1, Identified by the first 2 bits of the IGMP message as highlighted in green (figure 2), this usually denotes that the message has been send by a Multicast enabled router to discover membership groups on that specific interface. This message is normally sent to the “permanent group” address of 224.0.0.1, this is not the case within this packet, and therefore, this does not conform to the standard query. The IGMP payload does contain a multicast group address as highlighted in yellow (figure 2), this address is 240.0.3.146 and falls within the “E” Class range, as stated within RFC 3330, and is not currently used within the internet environment.

The use of the 240.0.3.146 address now becomes quite significant to achieve the DoS. As stated above, a router using the dense mode of operation will flood any queries that it has not previously received, to all interfaces. As this address is not used within the internet environment, any router that received this query will follow this rule, as it is guaranteed that they would not have received a request for the groups membership previously, this will give maximum dissemination with little effort.

There now comes another twist in the tail, these packets are crafted with a “C” class address, and not the expected “D” class address of 240.0.0.1. consiquently the attacker has quite cleverly told the router the destination subnet that this request is intended for. The router will now route the request to the required interface, and thus the target host, now the target host will attempt to respond to the query and fall into the loop error of the Land Attack.

The use of IGMP is an attempt to bypass any Firewall present, by assuming that the Firewall configuration will allow Multicast Management messages to pass through it. If this is the case, the attacker will have achieved his task and the target host will either crash or slow down, depending on the OS, due to the initialisation of a looping error. Although the example used only discusses one target IP, the packet capture that this example has been taken from contains 33 similar events, all with unique IP addresses. I assess that the attacker has assumed that the Router will have been patched against this type of attack; therefore, to perform the DOS, they will have to target the workstations themselves (Figure 4)

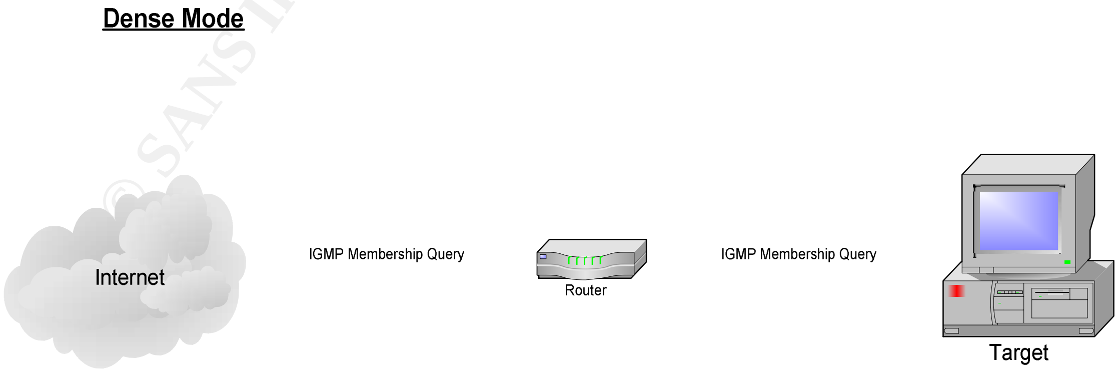


Figure 4.

6. Correlations

Maximum Security Fourth Edition.
Published by Sams (www.samspublishing.com).
Written by Anonymous, ISBN 0-672-32459-8

IP Routing Primer Plus
Published by Sams (www.samspublishing.com)
Written by Heather Osterloh

CNET News.com – Bug Threatens Net software, by Ben Heskett
http://news.com.com/2100-1001-205989.html

CISCO Security Advisory: TCP Loopback DoS Attack (land.c) and Cisco Devices Revision 5
http://www.cisco.com/warp/public/770/land-pub.shtml

RFC 1112, Host Extensions for IP Multicasting, Written by S. Deering Stanford University.
http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1112.html

RFC 3330, IPv4 Addresses
http://www.rfc-editor.org/rfc/rfc3330.txt

RFC 3171, IANA Guidelines for IPv4 Multicast Address Assignments.
http://www.faqs.org/rfcs/rfc3171.html
Written by Z. Albanna, K Almeroth, D. Meyer, M. Schipper

7. Evidence of active Targeting

There is evidence to support that the attacker is specifically targeting this network. The use of IP address 207.166.38.* suggests that the attacker is intent on DOS'ing this network.

8. Severity

The severity for this event would be based on the assessment that there are un-patched workstations/OS's within the network, which could be susceptible to this type of attack. Therefore, the following values will reflect this.

Severity = (criticality+lethality) – (System countermeasures + network countermeasures)

Therefore

Severity = (2 + 5) – (0 + 4)

Severity = 7 – 4 = 3

This would indicate that the threat of DOS to the network is high.

9. Defensive Recommendation

The recommendations that I would make to prevent this tool from exploiting a network would be as follows:

a. Install/maintain a stateful firewall, ensuring that only ports that are needed for interaction with the internet are open. Ensure IGMP is dropped on externally facing interfaces at the border gateway.

b. Check and maintain the patch level of the system, paying particular attention to critical machines.

c. Install/Maintain/monitor NIDS behind the firewall, consider HIDS on critical systems, this will identify any malicious activity.

d. Use IGMP Version 3, which supports Select Source Multicasting, this utilises a filtering system based on source IP address.

10. Test Question:

The presence of Land Attack is indicated by which unique criteria within the packet capture?

a. The TTL.
b. The IP Packet size
c. The source/destination IP address
d. The payload held within the packet

Answer C, the source and destination are always the same, but all should be considered due to the packet being crafted.

Assignment #2: Network Detects
Detect 3: RingZero

1. Source of Trace.

The raw packet capture used for this exercise was obtained from http://www.incidents.org/logs/Raw, and was selected from the 2002.9.19 capture. From the analysis of the data, I consider the network layout to be as detailed at figure 1. This conclusion is made as the source address and destination IP addresses are associated with the MAC address which falls within the range used by CISCO, and under standard network configuration the IP source will take on the MAC address of the last router the packet is passed through.

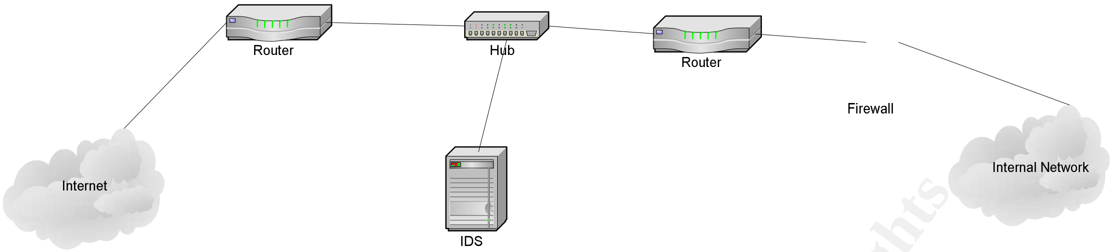


Figure 1

2. Detect was generated by:

The detection suite that I used for this exercise was as follows:

a. A default installation of the Snort 2.0 build sensor, with standard logging, alert file generation and rule sets. http://www.snort.org/dl/binaries/win32

b. Windump, http://windump.polito.it/install/default.htm

c. Ethereal 0.9.11, http://www.ethereal.com/download.html

d. Eagle X IDSCentre 1.1 RC4 Installation, http://www.engagesecurity.com/downloads

   (Note: The standard fit for this build is a Snort 2.1 engine, I changed this to a 2.0 engine due to a possible bug in the later version while replaying events (-r). The snort 2.1 did not generate the expected alerts when collated with the output of Windump and Ethereal).

The reasons for choosing this suite of tools, boils down to familiarity. The Snort 2.0 build was chosen as a base line sensor with which to test the output from the Eagle X build. The Eagle X IDS Centre was chosen to provide a user friendly tool for correlation of events, and to make use of the PHP facilities and potted queries, thus reducing the workload necessary on the volume of information generated. Both Windump and Ethereal were used to analyse the individual packet content, cross referring information to confirm the validity of my finding, and identifying any false positives.

Snort rules that Generated alerts:

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 3128 (msg:"SCAN Squid Proxy
attempt"; flags:S,12; classtype:attempted-recon; sid:618; rev:4;)
```

```
alert tcp $EXTERNAL_NET any -> $HOME_NET 8080 (msg:"SCAN Proxy \(8080\)
attempt"; flags:S,12; classtype:attempted-recon; sid:620; rev:3;)
```

Alerts Generated by Eagle X IDSCentre

Extracted from the Unique alert list.

Signature	Class	# EVT	#Unique			Start time	End time
[snort] SCAN Squid Proxy attempt	attempted-recon	3642 (49%)	1	2	1983	2002-10-19 09:29:00	2002-10-19 16:26:44
[snort] SCAN Proxy (8080) attempt	attempted-recon	3681 (49%)	1	2	1985	2002-10-19 09:29:02	2002-10-19 16:26:23

Extracted from Query Results by drilling down through "Traffic Profile by Protocol"

Signature	Time	Source IP	Dest IP	Prot
[snort] SCAN Squid Proxy attempt	2002-10-19 09:37:44	24.190.48.235:2545	32.245.123.86:3128	TCP
[snort] SCAN Proxy (8080) attempt	2002-10-19 09:37:44	24.190.48.235:2772	32.245.124.128:8080	TCP

Extracted from a Query Result on above ID # for event (not shown)

Meta

ID #	Time	Triggered Signature
2 - 6645	2002-10-19 09:37:44	[snort] SCAN Squid Proxy attempt

Sensor	name	interface	filter
	DEFIANT:[reading from a file]	[reading from a file]	*none*

Alert Group	*none*

IP

source addr	dest addr	Ver	Hdr Len	TOS	length	ID	flags	offset	TTL	chksum
24.190.48.235	32.245.123.86	4	5	0	44	21618	0	0	120	45647

FQDN	Source Name	Dest. Name
	Unable to resolve address	*Unable to resolve address*

Options	*none*

TCP

source port	dest port	R1	R0	URG	ACK	PSH	RST	SYN	FIN	seq #	ack	offset	res	window	urp	chksum
2545	3128							X		1501067715	0	6	0	8192	0	37559

Options		code	length	data
	#1	MSS	2	05B4

Payload	*none*

Meta

ID #	Time	Triggered Signature
2 - 6650	2002-10-19 09:37:44	[snort] SCAN Proxy (8080) attempt

Sensor	name	interface	filter
	DEFIANT:[reading from a file]	[reading from a file]	*none*

Alert Group	*none*

IP

source addr	dest addr	Ver	Hdr Len	TOS	length	ID	flags	offset	TTL	chksum
24.190.48.235	32.245.124.128	4	5	0	44	24178	0	0	120	43043

FQDN	Source Name	Dest. Name
	Unable to resolve address	*Unable to resolve address*

Options	*none*

TCP

source port	dest port	R 1	R 0	U R G	A C K	P S H	R S T	S Y N	F I N	seq #	ack	offset	res	window	urp	chksum
2772	8080							X		1503373268	0	6	0	8192	0	20508

Options		code	length	data
	#1	MSS	2	05B4

Payload	*none*

The above output is an HTML, user friendly, representation of the Hexadecimal/ASCII dump of the Snort capture.

Extracted from Snort 2.0 (Default Installation) Alerts file;

```
[**] SCAN Squid Proxy attempt [**]
10/19-09:30:04.826507 0:3:E3:D9:26:C0 -> 0:0:C:4:B2:33 type:0x800 len:0x3C
24.190.48.235:2405 -> 32.245.117.86:3128 TCP TTL:120 TOS:0x0 ID:56298 IpLen:20
DgmLen:44 DF
******S* Seq: 0x591D34DD  Ack: 0x0  Win: 0x2000  TcpLen: 24
TCP Options (1) => MSS: 1460
0x0000: 00 00 0C 04 B2 33 00 03 E3 D9 26 C0 08 00 45 00  .....3....&...E.
0x0010: 00 2C DB EA 40 00 78 06 30 D7 18 BE 30 EB 20 F5  .,..@.x.0...0. .
0x0020: 75 56 09 65 0C 38 59 1D 34 DD 00 00 00 00 60 02  uV.e.8Y.4.....`.
0x0030: 20 00 DE 84 00 00 02 04 05 B4 00 00             ...........
```

```
[**] SCAN Proxy (8080) attempt [**]
10/19-09:30:04.826507 0:3:E3:D9:26:C0 -> 0:0:C:4:B2:33 type:0x800 len:0x3C
24.190.48.235:2404 -> 32.245.117.86:8080 TCP TTL:120 TOS:0x0 ID:56042 IpLen:20
DgmLen:44 DF
******S* Seq: 0x591D34CE  Ack: 0x0  Win: 0x2000  TcpLen: 24
TCP Options (1) => MSS: 1460
0x0000: 00 00 0C 04 B2 33 00 03 E3 D9 26 C0 08 00 45 00  .....3....&...E.
0x0010: 00 2C DA EA 40 00 78 06 31 D7 18 BE 30 EB 20 F5  .,..@.x.1...0. .
0x0020: 75 56 09 64 1F 90 59 1D 34 CE 00 00 00 00 60 02  uV.d..Y.4.....`.
0x0030: 20 00 CB 3C 00 00 02 04 05 B4 00 00             ..<........
```

```
[**] SCAN Squid Proxy attempt [**]
10/19-09:37:35.586507 0:3:E3:D9:26:C0 -> 0:0:C:4:B2:33 type:0x800 len:0x3C
24.190.48.235:2773 -> 32.245.124.128:3128 TCP TTL:120 TOS:0x0 ID:60780
IpLen:20 DgmLen:44 DF
******S* Seq: 0x599BA7D6  Ack: 0x0  Win: 0x2000  TcpLen: 24
TCP Options (1) => MSS: 1460
0x0000: 00 00 0C 04 B2 33 00 03 E3 D9 26 C0 08 00 45 00  .....3....&...E.
0x0010: 00 2C ED 6C 40 00 78 06 19 29 18 BE 30 EB 20 F5  .,.l@.x..)..0. .
0x0020: 7C 80 0A D5 0C 38 59 9B A7 D6 00 00 00 00 60 02  |....8Y.......`.
0x0030: 20 00 63 71 00 00 02 04 05 B4 00 00             .cq........
```

```
[**] SCAN Proxy (8080) attempt [**]
10/19-09:37:35.586507 0:3:E3:D9:26:C0 -> 0:0:C:4:B2:33 type:0x800 len:0x3C
24.190.48.235:2772 -> 32.245.124.128:8080 TCP TTL:120 TOS:0x0 ID:60524
IpLen:20 DgmLen:44 DF
******S* Seq: 0x599BA7D4  Ack: 0x0  Win: 0x2000  TcpLen: 24
TCP Options (1) => MSS: 1460
```

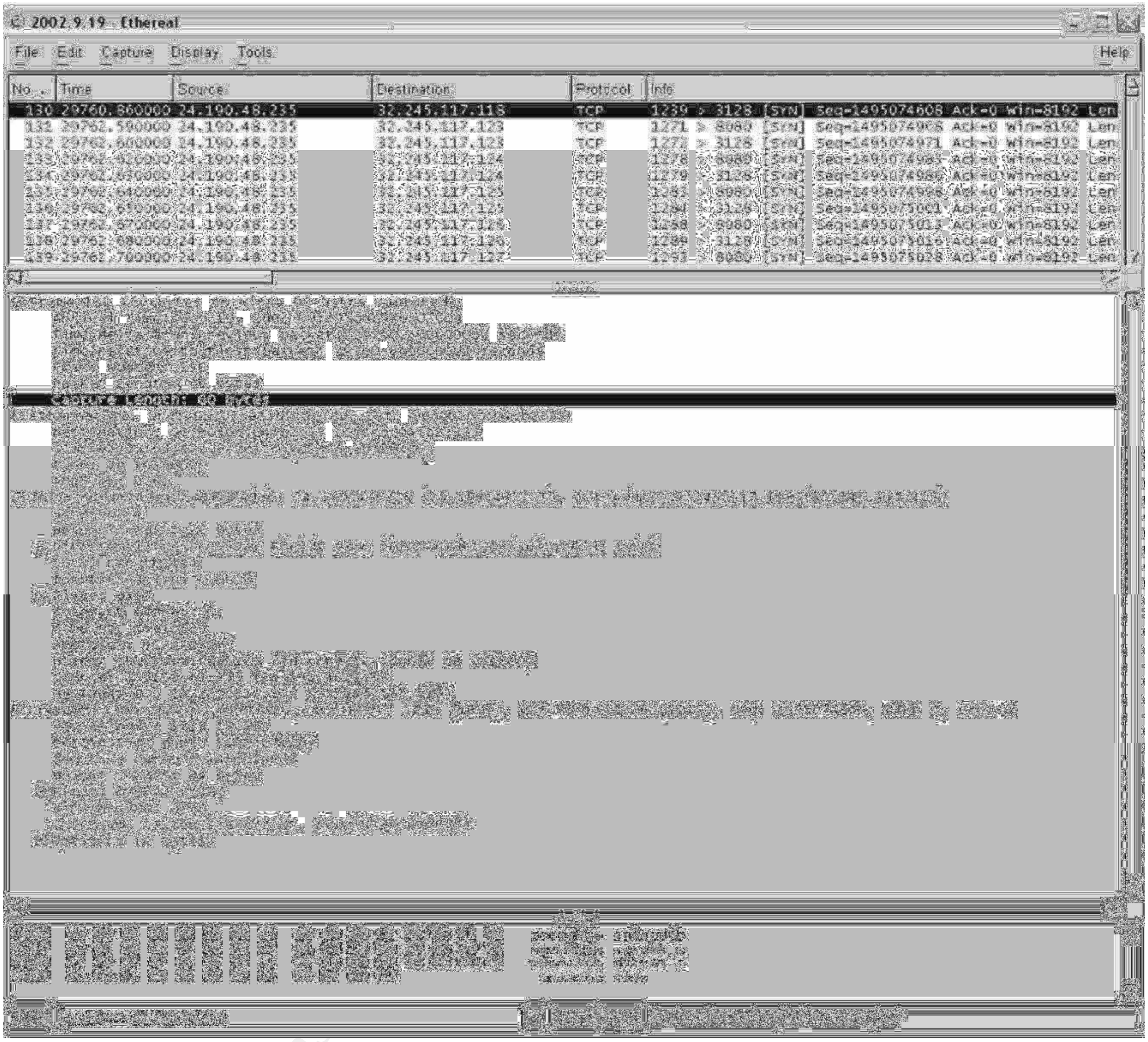
```
0x0000: 00 00 0C 04 B2 33 00 03 E3 D9 26 C0 08 00 45 00  .....3....&...E.
0x0010: 00 2C EC 6C 40 00 78 06 1A 29 18 BE 30 EB 20 F5  .,.l@.x..)..0. .
0x0020: 7C 80 0A D4 1F 90 59 9B A7 D4 00 00 00 00 60 02  |.....Y.......`.
0x0030: 20 00 50 1C 00 00 02 04 05 B4 00 00              .P.........
```

3. Probability the Source address was spoofed

The probability that this packet contains a spoofed source address is low. I have considered that this packet has been generated from outside the network, as the source MAC is associated with the allocation which falls within the range used by CISCO; therefore the packet has originated from the internet facing gateway. A full explanation of why this packet does not contain a spoofed IP address will follow under the Description of the attack section.

4. Description of the Attack

RingZero is a known Trojan, which exhibits similar patterns to the one seen within this the packets analysed. The main delivery system for this Trojan is via e-mail, and is identified as being attached to a Winsock Version Checker program, but can be attached to any Windows executable. The pure RingZero has 2 parts, an executable and a VXD part which is attached to the executable. When the attachment is run it detaches itself form the delivery file and writes two file to the system (\windows\system), Ring0.vxd and any one of the following; Telnet23.exe, explupd.exe, pct.exe or its.exe, the latter being the most common file found, a file a.exe may also be created.



```
No. . Time         Source         Destination     Protocol Info
  130 29760.860000 24.190.48.235  32.245.117.118  TCP      1239 > 3128 [SYN] Seq=1495074608 Ack=0 Win=8192 Len
  131 29762.590000 24.190.48.235  32.245.117.123  TCP      1271 > 8080 [SYN] Seq=1495074968 Ack=0 Win=8192 Len
  132 29762.600000 24.190.48.235  32.245.117.123  TCP      1272 > 3128 [SYN] Seq=1495074971 Ack=0 Win=8192 Len
  133 29762.620000 24.190.48.235  32.245.117.124  TCP      1278 > 8080 [SYN] Seq=1495074985 Ack=0 Win=8192 Len
  134 29762.630000 24.190.48.235  32.245.117.124  TCP      1279 > 3128 [SYN] Seq=1495074988 Ack=0 Win=8192 Len
  135 29762.640000 24.190.48.235  32.245.117.125  TCP      1281 > 8080 [SYN] Seq=1495074998 Ack=0 Win=8192 Len
  136 29762.650000 24.190.48.235  32.245.117.125  TCP      1284 > 3128 [SYN] Seq=1495075001 Ack=0 Win=8192 Len
  137 29762.670000 24.190.48.235  32.245.117.126  TCP      1288 > 8080 [SYN] Seq=1495075013 Ack=0 Win=8192 Len
  138 29762.680000 24.190.48.235  32.245.117.126  TCP      1289 > 3128 [SYN] Seq=1495075016 Ack=0 Win=8192 Len
  139 29762.700000 24.190.48.235  32.245.117.127  TCP      1293 > 8080 [SYN] Seq=1495075028 Ack=0 Win=8192 Len

Capture Length: 60 bytes
```

Figure 2

You will see form Figure 2 and the Snort Port Scan Log Extract that the ports targeted by this source address are 3128 and 8080, and that port 80 port scans have been dropped.

5. Attack Mechanism

Once a host has been infected it will perform a scan against a random IP range, this scan will climb the range sequentially as seen within the logs above. It will scan for active proxy servers on ports 80 (Common HTTP port), 8080 (common WWW PROXY SERVICES port) and 3128 (SQUID PROXY port). When an active proxy service is found, the Trojan will copy the IP information to its ITS.DAT file and send this information back to www.rusftpsearch.net (no longer active). Were a list of the infected hosts was stored under the heading “the biggest proxy host list ever”.

The following quote was found on the Network Associates Website (http://vil.nai.com/vil/content/v_10356.htm).

> One version of this Trojan writes the filenames "ITS.EXE", "PST.EXE" and also "RING0.VXD" within a distributed file "GETGR3_1.EXE". This program was a Trojanzed version of a game program which turned the mouse cursor into a pistol where the user could "shoot holes into the screen".

6. Correlations:

The Hunt for The RingZero Trojan; Written by John Green (jegreen@crosslink.net)

Symantec Security Response: RingZero.Trojan
http://www.symantec.com/avcenter/venc/data/ringzero.trojan.html

F-Secure Virus Descriptions: RingZero
http://www.f-secure.com/v-descs/ringzero.shtml

National Infrastructure Protection Centre: Advisory 99-024, RingZero Trojan Program
http://www.nipc.gov/warnings/advisories/1999/99-024.htm

Network Associates
http://vil.nai.com/vil/content/v_10356.htm

Pest Patrol: RingZero
http://www.pestpatrol.com/PestInfo/r/ringzero.asp

CIS Knowledge Base: The RingZero Trojan
http://www.unh.edu/cis-workstation/security/ringzero.html

The Hunt for RingZero (port 3128 scanning worm/Trojan) (From Andrew Daviel)
http://www.squid-cache.org/mail-archive/squid-users/199911/0482.html

7. Evidence of Active Targeting

This is not a targeted scan, it is more of the shotgun approach, the aim is to infect as many hosts as possible and retrieve information to compile a list of Proxy Servers, by distributing the scanning process to a large number of hosts. The use of this information could only be for malicious purposes, such as DoS attacks or to provide anonymous connection for Hackers/Crackers while performing malicious activity.

8. Severity

The severity for this event would be based on the assessment that there are Windows Based Workstation within the network which are running unpatched versions Outlook express/Outlook which will run the .exe file without interaction with the user. There are system countermeasures in place in the form of a maintained Anti Virus product, and the network countermeasure are a stateful firewall with a “deny all except that which is explicitly allowed” policy in force. Therefore, the following values will reflect this.

Severity = (criticality+lethality) – (System countermeasures + network countermeasures)

Therefore

Severity = (3 + 3) – (4 + 5)
Severity = 6 – 9 = -3

This would indicate that the threat to the network is low, and the scan is more of an annoyance that a threat. However, vigilance is still recommended.

9. Defensive Recommendation

The recommendations that I would make to prevent this tool from exploiting a network would be as follows:

a. Install/maintain a stateful firewall, ensuring that only ports that are needed for interaction with the Internet are open. Ensure anti spoofing is activated on the firewall this will drop all internal addresses that are generated from the external network.

b. Check and maintain the patch level of the system, paying particular attention to critical machines.

c. Install/maintain a system integrity checker, i.e. Tripwire, to highlight any changes that are made without authorisation.

d. Install/Maintain/monitor NIDS (Network Intrusion Detection System)behind the firewall, consider HIDS (Host Intrusion Detection System) on critical systems. This will identify any compromise that may take place due to a zero day event or firewall failure.

e. Configure your email server to block or remove email that contains file attachments that are commonly used to spread viruses, such as .vbs, .bat, .exe, .pif and .scr files

f. Maintain system auditing, check and monitor such event logs.

g. Install/maintain an Antivirus solution.

10. Test Question:

What was the main aim of the deployment of RingZero ?

a. To compromise as many hosts as possible.
b. Distributed Denial of Service (DDoS).
c. Produce a list of usable Proxy Servers to be exploited.
d. Promote the use of Proxy Servers as a valuable Security Tool

Answer C.

Assignment 3;
The Scenario “This is life Jim, but not as we know it!”

1. The Data

The following files were downloaded from the Intrusion.org site for analysis. The files selected range for the 19 – 23 Oct 03 inclusive, and represent a full 5 days of Alert, Scan and OOS data.

Alert	Scan	OOS
Alert.031019	Scan.031019	OOS_Reports_2003_10_19
Alert.031020	Scan.031020	OOS_Reports_2003_10_20
Alert.031021	Scan.031021	OOS_Reports_2003_10_21
Alert.031022	Scan.031022	OOS_Reports_2003_10_22
Alert.031023	Scan.031023	OOS_Reports_2003_10_23

Table 1. Data from Intrusion.org for analysis.

2. Executive Summary

The evidence held within the data above has indicated that this organisation is infested with Hackers with nothing more on their mind but malicious intent. Through careful analysis this statement can be tempered, as not all the activity within these logs is indicative of malicious activity.

The analysis has shown that the IDS rules need to be re-assessed and tuned to reflect the current usage of the internal network. This would reduce the amount of false positives being generated, and thus highlight any truly malicious activities that present themselves. This thought process must also be employed within the network itself, by identifying machines that are producing network noise and reducing their output, we directly decrease the bandwidth usage on the internal network and thus make it more efficient.

Although this report has indicated the presence of malicious activity, these activities require further investigation to confirm that they are as indicated. This type of investigation entails packet analysis which is outside the scope of the data provided. The types of activity can be described in one word “why”, i.e. why are internal machines scanning externally? Why are known Trojans active within the internal network? These are just some of the questions that have been investigated within this report.

3. Alert Signatures and Priorities

After collating the alert, Scan and OOS files into an Access database (annex a), a number of queries were produced and performed against the database, this facilitated the formulation of a number of tables which were then used to identify any potential malicious activities.

Table 2 highlights the alerts which have appeared within this reporting period and the relationship between the number of times a signatures appeared within the data. These aided in the initial prioritisation of the events, from this table a more granular prioritisation can be undertaken by utilising three factors (Table 3):

a. Number of events
b. Potential Malicious activity
c. Analysts Gut instinct

Alert Signature Count	
Signature	**CountOfSignature**
SMB Name Wildcard	199212
SMB C access	28546
MY.NET.30.4 activity	15606
EXPLOIT x86 NOOP	11563
connect to 515 from inside	7131
MY.NET.30.3 activity	5726
TCP SRC and DST outside network	4518
External RPC call	3266
High port 65535 tcp - possible Red Worm - traffic	3172
Possible trojan server activity	2009
ICMP SRC and DST outside network	1825
NMAP TCP ping!	752
SUNRPC highport access!	494
Null scan!	455
High port 65535 udp - possible Red Worm - traffic	438
[UMBC NIDS IRC Alert] IRC user /kill detected	342

Alert Signature Count

Signature	CountOfSignature
scan (Externally-based)	260
[UMBC NIDS IRC Alert] XDCC client detected attempting to IRC	182
FTP passwd attempt	105
[UMBC NIDS] External MiMail alert	103
Back Orifice	84
TFTP - Internal UDP connection to external tftp server	83
Incomplete Packet Fragments Discarded	74
Tiny Fragments - Possible Hostile Activity	62
[UMBC NIDS IRC Alert] Possible sdbot floodnet detected attempting to IRC	55
EXPLOIT x86 stealth noop	53
NETBIOS NT NULL session	51
DDOS shaft client to handler	38
[UMBC NIDS IRC Alert] Possible drone command detected.	37
EXPLOIT x86 setuid 0	27
EXPLOIT x86 setgid 0	26
EXPLOIT NTPDX buffer overflow	25
DDOS mstream client to handler	14
FTP DoS ftpd globbing	14
TFTP - Internal TCP connection to external tftp server	13
[UMBC NIDS IRC Alert] Possible Incoming XDCC Send Request Detected.	12
TFTP - External UDP connection to internal tftp server	11
Attempted Sun RPC high port access	10
RFB - Possible WinVNC - 010708-1	10
HelpDesk MY.NET.70.49 to External FTP	5
[UMBC NIDS IRC Alert] K\:line'd user detected	4
NIMDA - Attempt to execute cmd from campus host	4
[UMBC NIDS] Internal MSBlast Infection Request	3
External FTP to HelpDesk MY.NET.53.29	2
connect to 515 from outside	2
TFTP - External TCP connection to internal tftp server	2
Traffic from port 53 to port 123	2
External FTP to HelpDesk MY.NET.70.50	2
External FTP to HelpDesk MY.NET.70.49	2
Probable NMAP fingerprint attempt	2

Alert Signature Count	
Signature	**CountOfSignature**
Bugbear@MM virus in SMTP	1
[UMBC NIDS IRC Alert] Possible trojaned box detected attempting to IRC	1
IRC evil - running XDCC	1

Table 2. Alert Signature Count

Table 3 identified the signatures of interest for investigation after this further prioritization has taken place. These events will be covered in detail within the **Alert Analysis** section below:

Priority	Signatures	Number of events
1	SMB Name Wildcard	199212
2	Possible Trojan Server Activity	2009
3	TCP SRC and DST outside network	4518
4	External RPC Call	3266
5	SUNRPC highport access!	494

Table 3. Signature Prioritization

Table 4 shows the “Top 10 Takers”. It is the relationship of Source IP and the number of events that they have generated within the Alert logs. The pattern of event which has now been highlighted has identified one event above all others to be of significance. This event is the SMB Name Wildcard.

Alert Source IP Count	
Source IP	**CountOfSource IP**
MY.NET.80.51	115624
MY.NET.150.133	72067
MY.NET.162.41	7132
169.254.244.56	4279
MY.NET.29.2	3101
68.55.85.180	2934
193.114.70.169	2891
68.54.91.147	2743
MY.NET.84.224	1290
68.57.90.146	1251

Table 4. Alert Source IP Count

4. Scan Signatures and Prioritization

From the collated scan logs, a table was produced to highlight the top scan signatures being generated within the network (Table 5). The same process was used to investigate the "top talkers" and then cross refered to identify possible malicious activity (Table 6).

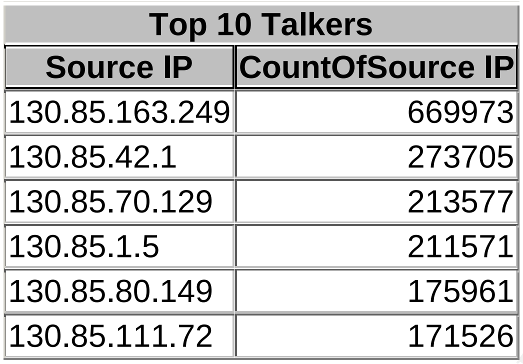
Scan Signature count

Signature	CountOfSignature
SYN scan (Externally-based)	8584047
UDP scan (Externally-based)	3108290
FIN scan (Externally-based)	2709
INVALIDACK scan (Externally-based)	2529
UNKNOWN scan (Externally-based)	1551
NULL scan (Externally-based)	347
NOACK scan (Externally-based)	176
VECNA scan (Externally-based)	52
scan (Externally-based)	8
XMAS scan (Externally-based)	7
SYNFIN scan (Externally-based)	4
22 scan (Externally-based)	3
NMAPID scan (Externally-based)	3
SPAU scan (Externally-based)	3
130.85.97.102:22321 scan (Externally-based)	2
-> scan (Externally-based)	2
FULLXMAS scan (Externally-based)	2
******S* scan (Externally-based)	1
23 scan (Externally-based)	1
216.52.121.228:53 scan (Externally-based)	1
130.85.80.51:1036 scan (Externally-based)	1
130.85.153.98:2861 scan (Externally-based)	1
130.85.1.3:62206 scan (Externally-based)	1

Table 5

Top 10 Talkers

Source IP	CountOfSource IP
130.85.1.3	2166933
130.85.70.154	1294187
130.85.163.107	966595
130.85.84.194	888185



Top 10 Talkers	
Source IP	CountOfSource IP
130.85.163.249	669973
130.85.42.1	273705
130.85.70.129	213577
130.85.1.5	211571
130.85.80.149	175961
130.85.111.72	171526

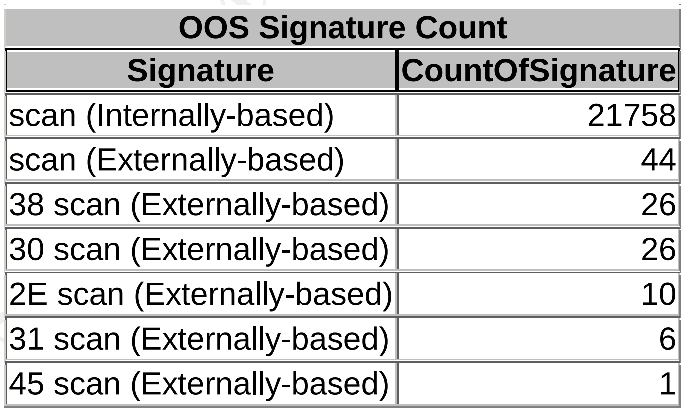
Table 6

This scanning activity has been prioritised due to the amount of activity performed by a single scan signature as indicated within Table 5. These events are analysed within the section entitled **Scans Analysis** below:

---

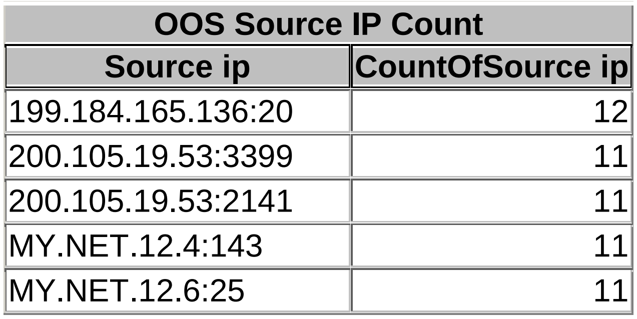
5. OOS Signatures and Prioritization

From the collated OOS logs, a table was produced to highlight the top scan signatures being generated within the network (Table 7). The same process was used to investigate the "top talkers" and then cross referred to identify possible malicious activity (Table 8).



OOS Signature Count	
Signature	CountOfSignature
scan (Internally-based)	21758
scan (Externally-based)	44
38 scan (Externally-based)	26
30 scan (Externally-based)	26
2E scan (Externally-based)	10
31 scan (Externally-based)	6
45 scan (Externally-based)	1

Table 7



OOS Source IP Count	
Source ip	CountOfSource ip
199.184.165.136:20	12
200.105.19.53:3399	11
200.105.19.53:2141	11
MY.NET.12.4:143	11
MY.NET.12.6:25	11

OOS Source IP Count	
Source ip	CountOfSource ip
195.208.238.143:51144	10
148.63.207.124:4405	10
148.63.207.124:3317	10
200.105.19.53:3931	10
66.28.62.36:20	10

Table 8

This OOS activity has been prioritised due to the amount of activity performed by a single host using a specific scan as indicated within Table 9 which highlights events that occur over 10 times, targeted against a unique IP address. These events are analysed within the section entitiled **OOS Analysis** below:

OOS Source IP Count Signature Target IP			
Source ip	CountOfSource ip	Signature	Target IP
61.175.193.250:14976	23	scan (Internally-based)	MY.NET.84.180
199.184.165.136:20	12	scan (Internally-based)	MY.NET.24.47
200.105.19.53:3399	11	scan (Internally-based)	MY.NET.150.133
200.105.19.53:2141	11	scan (Internally-based)	MY.NET.150.133

Table 9

Alert Analysis

6. SMB Name Wildcard

Priority: 1 **Number of events:** 199212

As the tables have indicated this signature above all others has been the most prolific, and was identified by:

a. Amount of events generated by this signature.
b. Number of Source IP address.

Alert Source IP count Sig		
SourceIP	CountOfSourceIP	Signature
MY.NET.80.51	115624	SMB Name Wildcard
MY.NET.150.133	72067	SMB Name Wildcard

Table 10.

As indicated by table 10, there have been two IP addresses generating this signature. The amount of traffic flow that these two machines generated is significant. Figure 1 has highlighted this fact, as the “top talker” accounted for 40% of the alert events, followed by the second place IP, which has generated just 25% as illustrated within Chart 1.

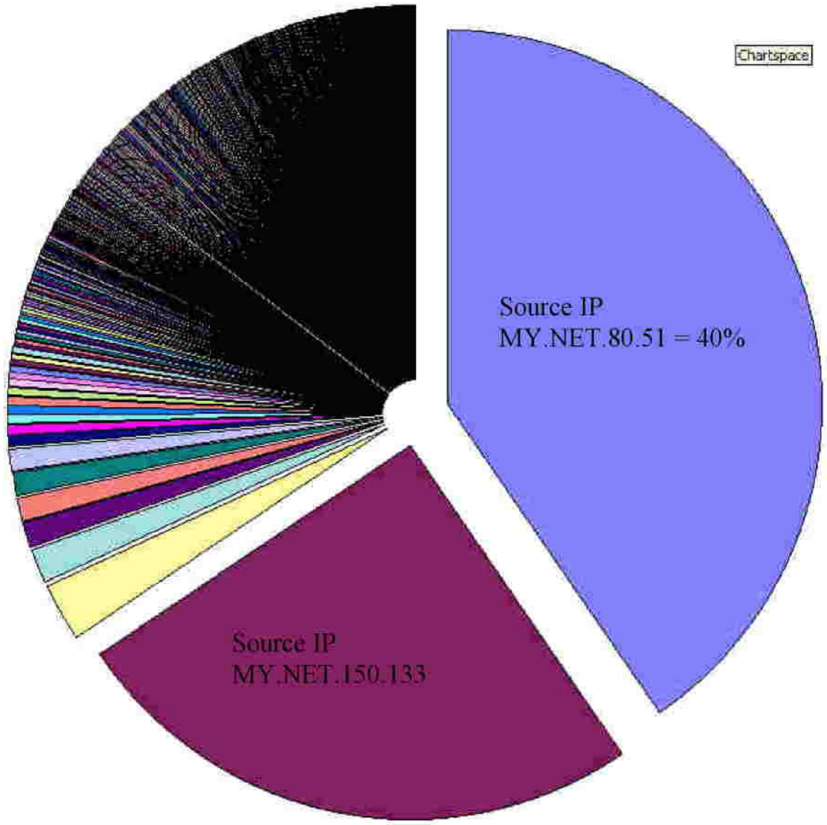


Chart 1

Analysis of this event found that MY.NET.80.51 generated the vast majority of this event (total of 115624). In most cases this event would be considered noise, as the alert can be generated through normal NetBIOS-NS traffic. This false positive would be easily identified as the source and destination port would both be 137. As illustrated below this is not the case, the source port of the packets generating this alert are all ephemeral. These ports were checked against the Treachery Unlimited lookup page (http://www.treachery.net/tools/ports/lookup.cgi), which showed that the only port which was linked to a known Trojan was 1035 – Multidropper. This tool has been described by Network Associates (http://www.nai.com/vil/content/v_99908.htm) as “ a trojan multidropper package designed to drop and execute other files on the target machine.” Therefore this event could not be considered normal traffic or network noise.

Alert Signature, Source IP & Source port Count			
Signature	Source IP	Source port	Count OfSource port
SMB Name Wildcard	MY.NET.80.51	1036	58149
SMB Name Wildcard	MY.NET.80.51	1035	57469
SMB Name Wildcard	MY.NET.150.133	3117	12173
SMB Name Wildcard	MY.NET.150.133	2128	10176
SMB Name Wildcard	MY.NET.150.133	1457	8824
SMB Name Wildcard	MY.NET.150.133	3895	7793

Table 11.

This event was collated with similar traffic generated within the preceding 5 days which also highlighted a number of scans using SMB, from the internal network addresses of MY.NET.162.118 and MY.Net.150.133, as well as MY.Net.80.51 as indicated above (Table 11).

The scan started on the 14th Oct 03, and continues though to the 23rd Oct. Each attempt at connection to the destination port 137 is attempted from an ephemeral port of 1025 and above, this is inconsistent with standard SMB traffic which will originate from 137 (NetBIOS-ns). The destination IP's all lie within a random subnet, e.g. 211.91.*.*, and ascending the subnet incrementally, until the subnet has been exhausted.

There has been much discussion on port 137 being used as a destination port for scanning activity within the security community's forums. A paper written by Bryce Alexander (May 10, 2000) http://www.sans.org/resources/idfaq/port_137.php, which discussed port 137 being used in this way, along with the relationship between the SMB Name Wildcard signature and this type of scan. Bryce Alexander also identifies that this tactic is being utilised by the "Script Kiddie" community, within published tools. Although traffic packet analysis techniques cannot be used on such alert logs, the pattern of this scan does not match that of the above tool described within this paper. This does not mean that a tool was not used to perform this task, just that it could not be easily identified form the data provided. A cross-reference of the latest prolific Worms and Viruses using the F-Secure website (www.f-secure.com), the Sophos website (www.sophos.com) and the Network Associates website (www.nai.com) was used to identify any worm or virus activity that may be related to this event, but no such activity was identified at the time of writing this document.

It is imperative that this event be investigated further. The source machines should be analysed and an in-depth packet analysis be undertaken to identify the true nature of the activity. This will either confirm the activity as malicious, identify the individual or individuals responsible for this activity/scan, or be assessed as benign. Both alternatives will facilitate the same result of stopping this activity, 1. by undertaking disciplinary action against the malicious users, or 2. by reconfiguring the signatures to a more realistic level to reflect the network configuration, as described by Max Vision (http://archives.neohapsis.com/archives/snort/2000-01/0220.html).

7. Possible Trojan Server Activity

Priority: 2 **Number of events:** 2009

No network administrator likes to admit that they have been compromised by a Trojan, but within today's networks this is a real and considerable threat. Although most network defences preclude much of this activity, some slip the net. This signature has been identified as coming from a number of machines both inside and outside the networks, most of which are false positives, as the communications have been initiated from a known port (i.e. port 25 SMTP) and are considered as a normal connection request. However Table 12 illustrates that two machines stand out above the rest.

Alert Source Port Count Target IP					
Source port	**CountOfSource port**	**Source IP**	**Signature**	**Target IP**	**Date**
27374	553	200.163.61.175	Possible trojan server activity	MY.NET.163.249	23
6667	402	MY.NET.163.249	Possible trojan server activity	200.163.61.175	23

Table 12

Apart from the obvious high number of events generated by each source port, it is not immediately apparent what the real significance of this table may be. On closer investigation it is revealed that the two machines, are in-fact, communicating with each other by using a know Trojan. The source ports were investigated using the Treachery website as detailed above, and a cross reference was found between these two ports, this was SubSeven 2.1.4 Defcon 8. Although we are seeing the ports listed as a source ports, these are actually replies to stimulus received form the target IP as illustrated in figure 1

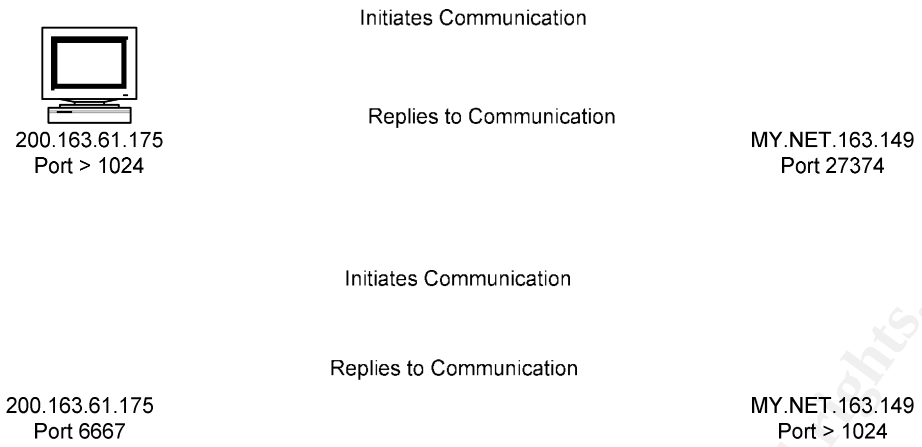


Figure 1.

The event is conducting the following:

a. 200.163.61.175 initiates a connection to MY.NET.163.149, which is listening on port 27374 for an incoming connection (SubSeven)
b. MY.NET.163.149 initiates a connection to 200.163.61.175, which is listening on port 6667 for an incoming connection (SubSeven)

These events all fall within the same time bracket on the 23rd Oct, from approx 0100 – 0300 (24 hours clock used).

It is my assessment that SubSeven is being used to provide remote access to assets held on both workstations. The intent of this activity may not be malicious, but the use of such tools as SubSeven to provide this type of "network share" is considered to be a serious security risk.

---

8. TCP SRC and DST outside network

Priority: 3 Number of events: 4518

This signature itself indicated the danger of the source and destination IP addresses falling outside the internal network. The indications are, that the internal network is being utilised as a hop off point, or as more commonly known a proxy, to contact other systems, possible to mask the identity of the source. The only reason that an individual would need to mask their identity would be, if they were involved in malicious or illegal activities.

Although a number of machines have been identified as being responsible for this traffic; 169.254.244.56 (linklocal IP address) accounted for 94.71% of the total traffic generated. As stated within RFC 3330, this IP address is obtained using auto-configuration when a host is unable to find a DHCP server. If a host is unable to find a DHCP server this would indicate that it does not have access to the internet, as an IP address would be issued by the ISP DHCP server upon connection. This IP would only be visible on an internal network, therefore this activity is assessed as a false positive or network noise. As this source IP has targeted five other machines outside the network, this would suggest that it has been configured to use a proxy server or NAT (name address translation) to gain access to the internet. The brunt of this traffic is being directed at two machines in particular, these are:

a. 211.91.144.72, within the range register to APNIC (Asian Pacific Network Information Centre) 1420 alerts, Trace Route indicated that the possible location was Beijing.

b. 218.16.124.131, within the range register to APNIC, 2851 alerts, Trace Route indicated that the possible location was Guangzhou.

c. 202.114.102.130, within the range register to APNIC, 3 alerts, Trace Route indicated that the possible location was Wuhan.

d. 205.188.75.94, within the range register to America Online, 2 alerts.

e. 208.16.124.131, within the range register to Sprint (Broadband Direct ISP), 3 alerts.

The registrants of the above IP's were investigated using Name Space (http://name.space.xs2.net/search/) which check the listings again IANA. The similarity between the IP address at b and e, could indicate an error while typing the IP address, or could just be coincidental.

The source IP addresses 68.55.0.64, which falls within the range used by Comcast Cable Communications, Inc Baltimore, has been successfully utilising this activity. The total number of events that this IP generated was 78 within this 5 day period; all targeting IP address outside the network as indicated below. Although this activity has not generated a high number of alerts in comparison with other alerts; it is recommended that this matter be investigated further using packet analysis techniques, this will identify the nature of this activity as either malicious or benign.

Date	Signature	Source IP	Target IP	CountTarget IP
23	TCP SRC and DST outside network	68.55.0.64	137.99.138.11	1
23	TCP SRC and DST outside network	68.55.0.64	24.126.194.159	1
23	TCP SRC and DST outside network	68.55.0.64	128.239.211.236	1
23	TCP SRC and DST outside network	68.55.0.64	24.126.241.187	1
23	TCP SRC and DST outside network	68.55.0.64	24.30.231.118	1
23	TCP SRC and DST outside network	68.55.0.64	24.34.13.155	1
23	TCP SRC and DST outside network	68.55.0.64	66.108.116.237	2
23	TCP SRC and DST outside network	68.55.0.64	66.69.108.201	1
23	TCP SRC and DST outside network	68.55.0.64	66.91.101.90	1
23	TCP SRC and DST outside network	68.55.0.64	67.166.83.36	1
21	TCP SRC and DST outside network	68.55.0.64	24.95.15.137	1
21	TCP SRC and DST outside network	68.55.0.64	24.74.59.20	1
21	TCP SRC and DST outside network	68.55.0.64	207.172.166.71	1
21	TCP SRC and DST outside network	68.55.0.64	130.18.200.214	1

21	TCP SRC and DST outside network	68.55.0.64	130.13.163.150	1
21	TCP SRC and DST outside network	68.55.0.64	130.13.135.211	1
21	TCP SRC and DST outside network	68.55.0.64	12.216.194.213	1
21	TCP SRC and DST outside network	68.55.0.64	12.208.42.130	1
21	TCP SRC and DST outside network	68.55.0.64	66.41.99.230	1
22	TCP SRC and DST outside network	68.55.0.64	24.33.45.239	1
22	TCP SRC and DST outside network	68.55.0.64	24.34.158.39	2
22	TCP SRC and DST outside network	68.55.0.64	24.34.83.87	2
22	TCP SRC and DST outside network	68.55.0.64	65.26.184.249	2
22	TCP SRC and DST outside network	68.55.0.64	65.28.242.174	1
22	TCP SRC and DST outside network	68.55.0.64	65.29.144.28	1
22	TCP SRC and DST outside network	68.55.0.64	66.177.204.29	2
22	TCP SRC and DST outside network	68.55.0.64	66.56.104.223	1
22	TCP SRC and DST outside network	68.55.0.64	67.164.82.152	2
22	TCP SRC and DST outside network	68.55.0.64	66.65.141.179	1
22	TCP SRC and DST outside network	68.55.0.64	24.50.242.181	3
22	TCP SRC and DST outside network	68.55.0.64	66.57.253.33	2
22	TCP SRC and DST outside network	68.55.0.64	128.84.0.158	2
22	TCP SRC and DST outside network	68.55.0.64	65.26.193.116	1
22	TCP SRC and DST outside network	68.55.0.64	12.208.42.130	1
22	TCP SRC and DST outside network	68.55.0.64	24.29.116.208	1
22	TCP SRC and DST outside network	68.55.0.64	130.111.54.61	2
22	TCP SRC and DST outside network	68.55.0.64	130.13.131.217	1
22	TCP SRC and DST outside network	68.55.0.64	130.13.152.5	1
22	TCP SRC and DST outside network	68.55.0.64	130.13.163.150	1
22	TCP SRC and DST outside network	68.55.0.64	130.74.201.63	2
22	TCP SRC and DST outside network	68.55.0.64	165.123.129.46	1
22	TCP SRC and DST outside network	68.55.0.64	168.215.141.202	1
22	TCP SRC and DST outside network	68.55.0.64	207.172.166.71	1
22	TCP SRC and DST outside network	68.55.0.64	208.177.202.80	2
22	TCP SRC and DST outside network	68.55.0.64	24.136.149.172	1
22	TCP SRC and DST outside network	68.55.0.64	152.30.98.227	1
20	TCP SRC and DST outside network	68.55.0.64	66.57.142.41	1
20	TCP SRC and DST outside network	68.55.0.64	24.25.104.196	1
20	TCP SRC and DST outside network	68.55.0.64	141.213.187.198	1
20	TCP SRC and DST outside network	68.55.0.64	24.159.72.192	1
20	TCP SRC and DST outside network	68.55.0.64	24.228.57.201	2
20	TCP SRC and DST outside network	68.55.0.64	24.242.61.151	1
19	TCP SRC and DST outside network	68.55.0.64	152.19.202.164	1
20	TCP SRC and DST outside network	68.55.0.64	24.28.137.136	1
20	TCP SRC and DST outside network	68.55.0.64	24.44.211.1	2
20	TCP SRC and DST outside network	68.55.0.64	24.88.112.91	1
20	TCP SRC and DST outside network	68.55.0.64	128.119.74.129	1
20	TCP SRC and DST outside network	68.55.0.64	24.91.79.100	1
20	TCP SRC and DST outside network	68.55.0.64	130.111.151.148	1
20	TCP SRC and DST outside network	68.55.0.64	67.160.239.196	1
20	TCP SRC and DST outside network	68.55.0.64	24.88.200.15	1
19	TCP SRC and DST outside network	68.55.0.64	24.33.5.145	1
20	TCP SRC and DST outside network	68.55.0.64	152.19.185.224	1

9. External RPC Call

Priority: 4 Number of events: 3266

Within the current climate of the MS03-026 vulnerability that exploits RPC DCOM and the release of the Blaster and Nachi worms, both of which utilise this vulnerability to compromise un-patched machines. This signature is assessed as a higher priority. Although this signature alone cannot be relied upon as a true reflection of the possibility of infection, it can be used as a rule of thumb to indicate whether or not an administrator has employed good security practices.

This event was a directed attack at the internal network, which originated from a single IP address of 193.114.70.169 (falls within the RIPE network Coordination centre's range) on the 23rd Oct between 04:40 to 19:04. By cross referring this signature and that of the TFTP based connections to an external source, this would give an indication of the possibility of infected machines within the network. However, as there is no attempted TFTP connection to this specific IP address which would indicate the downloading of the payload of one of the above worms, this would suggest that there are no vulnerable/exploited machines within the internal network.

10. SUNRPC highport access!

Priority: 5 Number of events: 494

The analysis of this signature shows that 139 of the 494 connections reported were considered to be false positives, due to the legitimate use of port 32771 as an ephemeral port. This traffic which originated from a number of IP addresses outside the network are replies to legitimate requests for services, i.e. port 80 (HTTP web traffic), Port 25 (SMTP mail traffic) and port 32778 (SunRPC traffic).

The remaining 355 events that originated from 128.183.16.143 (Listed as within the range used by NASA) and targeted against three IP addresses within the internal network:

a. MY.NET.97.154
b. MY.NET.97.246
c. MY.NET.97.96

This activity requires further investigation as these events originated from the source port 22, this port has been linked to the following services by Treachery website:

a. SSH Remote Login Protocol
b. PCAnywere (Application)
c. Adore SSHD (Trojan)
d. Shaft (Trojan)

Both c and d within this list immediately highlight themselves due to the association of known Trojans, and must not be ruled out of the analysis. However, this could also indicate the use of one of the legitimate application as listed at a and b. Within this context the use of the word "legitimate" is an assumptions that these services/applications are allowed within this network. It would be necessary to read the Security Policy documentation to assess the validity of this assumption.

PCAnywere can be considered by the user as a useful tool, as it provides a simple interface so that workstations can easily synchronise/share information. It also provides a remote (graphical) console so that the remote user can have full control over the host, as if he were sitting in front of it. Therefore this application could just as easily be used for malicious purposes and in this context be considered a Trojan. This use of legitimate software application for malicious purposes is a real threat to any network and must be closely monitored and comprehensively covered within the Security policy documentation.

Scanning Analysis

11. SYN Scan Externally Based

SYN Scan Externally Based			
Source IP	Target IP	Target Port	No. of events
130.85.70.154	130.84.*.*	135	1294187
130.85.163.107	131.155.*.*	135	966595
130.85.84.194	130.134.*.*	135	888185
130.85.163.249	95.148.*.*	135	669973
130.85.42.1	68.33.*.*	135	273705
130.85.70.129	130.87.*.*	135	213577
130.85.80.149	134.173.*.*	135	175961
130.85.111.72	45.133.*.*	135	171526

Table 13

Table 13 is a collation of the top 10 talker from the Scan data. It highlights the amount of scans that were being undertaken on the internet, for vulnerable machines which have not been patched against the RPC DCOM vulnerability (MS03-026). This activity is indicative of the MSBlast (Blaster,Lovesan) family of worms and suggests that the above source IP addresses have been infected with this worm or one of the many variants as listed at the F-Secure website (www.f-secure.com).

The following has been extract from the F-Secure website (http://www.f-secure.com/v-descs/msblast.shtml) which explains the activity above:

"**Spreading algorithm**

> The worm uses a sequential scanning algorithm with random starting points. The algorithm has a mode when it favors networks surrounding the infected host.
>
> An IP address has a following structure: A.B.C.D
>
> First the worm fetches the IP address of the infected host and puts it into the variables above.
>
> Based on a random number between 1 and 20 either the hosts IP is used as a basis of scanning or a totally random IP is generated.
>
> If random number is greater or equal to 12 the host IP is used. In this case if C is greater then 20 the worm subtracts 20 from it. D is always set to 0.
>
> If the worm chooses to use a totally random start IP it generates A B and C from random numbers:
>
> A from 1 to 254
> B from 0 to 253
> C from 0 to 253
> D is always 0
>
> Using these base addresses Lovsan starts to scan for vulnerable hosts. The algorithm scans 20 hosts at a time, the targets are successive IP address starting from the base address. The worm tries to connect to port 135 on all the 20 hosts and check if the connection is successful. In that case Lovsan uses one of many different DCOM exploits to infiltrate the host. There are two hardcoded values in the exploit which are randomly chosen. These values make the exploit work on either Windows 2000 or Windows XP systems. When the exploit starts on the remote machine it opens a shell through which the worm copies itself to the host using TFTP (Trivial File Transfer Protocol). The client for FTPS comes with Windows 2000/XP systems and the worm has a built-in TFTP server. After the worm is copied to the remote host it is started there through the shell. “

It is recommended that the administrator checks the patch level of all Windows NT based machines within the network against the current recommendation of Microsoft. The installation and maintenance of an Anti Virus solution is also recommended to detect and quarantine such activity. This will prevent infection from the above worms.

---

12. UDP Scan Externally Based

Source IP:	130.85.1.3
Target IP:	Multiple Externally Based IP addresses
Number of events:	2164273
Classification:	Internet Noise

Source IP:	130.85.1.5
Target IP:	Multiple Externally Based IP addresses
Number of events:	2164273
Classification:	Internet Noise

This signature can be considered network noise, as the target port in each case is 53, which equates to DNS activity. A straw poll of the target IP revealed that in each case the target in a DNS server. Therefore, this event is classed as a false positive and assessed as normal DNS traffic. Due to the amount of UDP Scan traffic being generated from both these machines, and because this traffic equates to 99.8% of their total output, I would assess that these machines are primary or secondary DNS servers within their networks.

OOS Analysis

13. Scan internally based

Source IP:	61.175.193.250
Source Port:	14976
Target IP:	MY.NET.84.180
Target Port:	6883 (Delta Source Darkstar Trojan)
Number of Events:	23

As the above summery indicates, the source IP has initiated a connection to the target IP and requested a specific port each time. By using the Treachery website, this port was identified as a possible known Trojan tool, Delta Source Darkstar. As the source IP has repeatedly requested this port, there is a high possibility that this Trojan is active on this target, and further investigation is necessary.

14. Scan internally based

Source IP:	199.184.165.136
Source Port:	20
Target IP:	MY.NET.24.47
Target Port:	>1024
Number of Events:	12

The above summery indicates that the source IP has originated 12 connections to the Target from source port 20. However, it is assessed that this connection has been originated from the target IP who has requested an FTP connection on port 21, and that this is the data being passed back to the target from port 20. The source IP address has been identified by using the Name Space website as belonging to RNC Corporation.

An FTP connection was requested to this IP address, and successful FTP connection was initiated that allowed anonymous login. A readme file was present on the initial browser screen, and indicated that this was the XEmacs Development Group ftp archive. A google search confirmed that this was the development site for the XEmacs

Editing Environment, which is an open source tool with Multi-platform, Multi-Language and Multi-OS capabilities. Therefore, this activity can be considered non-malicious and assessed as a false positive.

15. Scan internally based

Source IP:	200.105.19.53
Source port:	2141/3399
Target IP:	MY.NET.150.133
Target Port:	1214
Number of Events:	11/11

The above summary indicates that the source IP address initiated 22 contacts to the Target machine. The target port has been identified as that being used by Kazar, a file sharing utility; use to distribute pirated copies of music and films. Kazar has a large number of security vulnerabilities and is considered to be a major security risk within any network.

The target machine should be investigated to assess whether the Kazar utility has been installed on the machine.

By cross referring this IP with the IP addresses within the Alert logs, it has been highlighted that the target IP has been significantly involved with the SMB Wildcard Alert which appeared earlier in this report. It has been identified that the target IP initiated 74 contacts with the source machine, which generated the SMB Wildcard Alert. This strengthens the need to investigate these signatures further to dispel the coincidence or prove it as a malicious activity.

16. Defensive Recommendations

a. Deploy and maintain a stateful firewall which will track activity originating from within the internal network, and block activity that has originated from the external network (unless specifically allowed), thus preventing the system being used as a proxy.

b. Assess the state of the IDS signatures, and tune them (if necessary) to reflect the current legitimate activities being undertaken within the internal network. Consider setting trusts between source and target IP's if the activity is classed as normal by the users and authorised by the Administrator. Check that any IDS system is updated with the latest signature packs. Ensure that the IDS systems employed within the network have ability to capture and store packet information which can be used to assist in further investigations of malicious activities.

c. Ensure that there is a maintained Antivirus application active on the internal machines to prevent/detect any malicious application such as Trojan being deployed.

d. Assess whether the use of applications such as subseven (known Trojan) and PCAnyware are necessary within the network, and consider alternative methods of information sharing.

e. Ensure that all machines are updated with the latest service packs relevant to the current operating system, along with the latest available patch updates.

f. Educate the users on the security implication of installing unauthorised applications/tools on the internal network.

17. References

Treachery Unlimited Website:
http://www.treachery.net/tools/ports/lookup.cgi

Network Associates Website:
http://www.nai.com/vil/content/v_99908.htm

White paper, written by Bryce Alexander (May 10, 2000)
http://www.sans.org/resources/idfaq/port_137.php

F-Secure website
www.f-secure.com

Sophos website
www.sophos.com

Network Associates website
www.nai.com

White paper, writen by Max Vision
http://archives.neohapsis.com/archives/snort/2000-01/0220.html

Name Space Website
http://name.space.xs2.net/search/

F-Secure website, Description of the Blaster Worm
http://www.f-secure.com/v-descs/msblast.shtml

Annex A: Tools Used to complete this paper

The analysis was performed on the following OS's, and utilising the following tools:

- Microsoft Windows XP Professional (Dell Latitude 840 Laptop)
- Microsoft Windows XP Home (Acer workstation)
- Mandrake 9 (Elonex workstation)
- Microsoft Word XP Pro Edition
- Microsoft Excel XP Pro Edition
- Microsoft Access XP Pro Edition
- ActiveState ActivePerl 5.6.1.630
- SnortSnarf
- WinSnort2HTML
- Google Search Engine
- The SANS Institue
- Snort Signatures Database
- Treachary Website
- Name Space Website
- Whitehats Arach NIDS Database
- Wingrep

I also utilised the following Perl Script:

```
#!/cygdrive/c/Perl/bin/perl.exe -w
# Name: csv.pl
# Reads in a Snort -A Fast style alert log which for some
# reason wasn't generated as CSV, and make it as such.
#
# Usage: csv.pl infile [outfile]
unless ($ARGV[0]) {
print "Need an input file!\n";
die "(Hint: go to http://www.research.umbc.edu/~andy and get one)\n";
}
unless ($ARGV[1]) {
$outfile = "$ARGV[0].csv";
} else {
$outfile = "$ARGV[1]";
}
open(INFILE,"$ARGV[0]") || die "Can't open $ARGV[0] for reading!\n";
open(OUTFILE,">$outfile") || die "Can't open $ARGV[1] for writing!\n";
print "Transforming $ARGV[0] into $outfile.\n";
print "Just a moment.";
@calendar=qw(Jan Feb Mar Apr May Jun Jul Aug Sep Oct Nov Dec);
while (<INFILE>) {
next unless /(\w{1,3}\.){2}(\d{1,3}\.\d{1,3})/; # Skip lines missing IPv4 IPs.
next if /spp_portscan/; # Skip portscan notifications.
chomp;
if (/ \[\*\*\] /) { # Alert report.
($date_and_time,$alert,$src_and_dst) = split(/\s+\[\*\*\]\s/);
($date,$time) = split(/-/,$date_and_time);
($month_number,$day) = split(/\//,$date);
$month = $calendar[$month_number-1];
($src,$dst) = split(/\s-\>\s/,$src_and_dst);
($src_ip,$src_port) = split(/:/,$src);
($dst_ip,$dst_port) = split(/:/,$dst);
$snort_entry="ALERT" ;
} else { # Scan report.
($month,$day,$time,$src,$arrow,$dst,$alert,$flags) = split;
undef $arrow;
($src_ip,$src_port) = split(/:/,$src);
$alert = "$alert scan (Internally-based)" if $src_ip =~ /^MY\.NET/;
$alert = "$alert scan (Externally-based)" unless $src_ip =~ /^MY\.NET/;
($dst_ip,$dst_port) = split(/:/,$dst);
$snort_entry="SCAN" ;
}
print OUTFILE "$snort_entry,";
print OUTFILE "$month,$day,$time,$alert,";
print OUTFILE "$src_ip,";
print OUTFILE "$src_port" if $src_port;
print OUTFILE "None" unless $src_port;
print OUTFILE ",";
print OUTFILE "$dst_ip";
print OUTFILE ",";
print OUTFILE "$dst_port" if $dst_port;
print OUTFILE "," if $flags;
print OUTFILE "None," unless $dst_port;
print OUTFILE "$flags" if $flags;
print OUTFILE "\n";
$happydots++;
print "." if $happydots % 100 == 0; # if $happydots == 100;
print "Just a moment." if $happydots % 46600 == 0;
}
```

The tools and OS's were used to correlate the large amount of information contained within the 3 file types. I used the Mandrake 9 OS to "cat" the like files together (cat file1 file2 file3 (etc) > Filecollated).

The collated file was ported back to the Windows system where the Perl Script above (obtained from the GIAC paper of Tod A. Beardsley (Thanks)), was used to produce a .csv file of each type.

The .csv files were then imported into a Access database (total size 1.99 GB) where a number of queries were generated to analyse the information.

An attempt was made to use SnortSnarf to assist in this analysis. This process chewed up the memory on both my Dell Laptop and my Acer workstation. It was at this point that I turned to the resources of my employer and attempted to run the process on a Xeon machine with 2 GB of RAM which again did not prove fruitful, so I gave its up as a bad idea. The configuration of the SnortSnarf script makes it unsuitable to perform its task on such a large amount of information, but I had fun trying and learned a few things in the process. Winsnort2html was also trialed and discarded for similar reasons.