Global Information Assurance Certification Paper

GCIA Practical for
SANS Darling Harbour
Submitted by Michael Wilkinson

GCIA Practical version 3.0

Assignment 1 – Describe the state of Intrusion Detection 1
Evaluating Network Intrusion Detection Systems 1
Definition of a Intrusion Detection System 1
Published IDS evaluations 1
Intrusion Detection System Comparisons 2
IDS Evaluation Methodologies 4
Criteria for Evaluating Network Intrusion Detection Systems 5
Measurement Criteria 5
Conclusion 8
References: 9
Assignment 2 - Network Detects 10
Trace 1 - Attempted exploit of Windows file sharing, by automated application. 10
Trace 2 - Netlogon response to private address 18
Trace 3 - IIS Vulnerability scan 21
Trace 4 – Packet from the broadcast address 29
Trace 5 - Traffic to the loopback address 127.0.0.1 31
Assignment 3 – Analyse This 34
Executive Summary 34
Overview 34
Comments on data collection devices 35
Analysis of alerts 36
Top Talkers 36
Activity over time 37
Analysis of Alert types 38
SMB Name wildcard 43
ICMP Echo Request L3retriever Ping 44
spp_http_decode: CGI Null Byte attack detected 46
High port 65535 UDP - possible Red Worm - traffic & High port 65535 TCP - possible Red Worm - traffic 47
INFO MSN IM Chat Data 48
Watchlist 000220 IL-ISDNNET-990517 48
ICMP echo request Nmap or HPing2 49
ICMP: 49
Web server Alerts 50
Ftp alerts 55
MYPARTY - possible My Party infection 525 56
Incomplete packet fragments discarded 179 56
INFO alerts 56
MISC Alerts 60
Exploit alerts 62
SCANS 64
Out of Scope Traffic 70
Systems for further investigation 70
Recommendations: 72
Method of analysis 73
References 77

Assignment 1 – Describe the state of Intrusion Detection

Evaluating Network Intrusion Detection Systems

As computer systems and the Internet have grown in size, complexity and usage the demands placed upon those responsible for ensuring the continued operation and security of these systems has also grown. This has lead to a demand for automated systems for detecting malicious activity on both individual hosts and networks. In line with our capitalistic society where a demand exists suppliers will seek to meet that demand. This has lead to the development of a range of Intrusion Detection Systems. Some of these systems are available as free open source applications, while others are offered as commercial products. As a result any organisation considering implementing a IDS has a range of options available. The aim of this document is to outline various critera that can be used to evaluate Network Intrusion Detection Systems.

It should be noted that any organisation seeking to implement a IDS is likely to have their own needs and requirements from the system. The environment in which the IDS will be implemented is likely to vary from one situation to another, as is the availability of staff, funding and other essential resources. The aim of this document is to provide a review of various methods that can be adapted to meet the needs of any organisation.

Definition of a Intrusion Detection System

A Intrusion detection system is generally considered to be any system designed to detect attempts compromise the integrity, confidentiality or availability of the protected network and associated computer systems. A Network Intrusion Detection System (NIDS) aims to detect attempted compromises by monitoring network traffic for indications that a attempted compromise is in progress, or a internal system is behaving in a manner which indicates it may already be compromised. A host based IDS (HIDS) monitors a single system for signs of compromise.

A important point raised by Ranum (1998) is that a Intrusion Detection System should only report intrusions that either will successfully compromise the target system, or have not been seen before. Thus attempts to exploit a known Windows 2000 vulnerability on a Solaris system should not cause a IDS to generate a alert (this event should still be logged). Currently most available IDSs do not provide this level of functionality. This is one example of a measure that should be used for evaluation of IDSs.

Published IDS evaluations

At this point it would be worthwhile to outline some of the evaluations that have been published, and discuss the difference between a evaluation and a comparison. For the purposes of this paper a evaluation a considered to be a determination of the level to which a particular IDS meets specified performance targets. A comparison is considered to be a process of 'comparing' two or more systems in order to differentiate between them. It is proposed that a organisation intending to implement a IDS will

increase the likelihood of a successful implementation by establishing their own requirements of a IDS and then evaluating the available options to determine the level to which these requirements are met. This is the alternative to conducting a comparison of the available systems and then selecting the one which appears to be the 'best'.

The majority of published documents claiming to evaluate IDSs are conducted as comparisons, rather than evaluations. These documents serve as a useful starting point for any organisation considering the implementation of a IDS, however they may not prove sufficient, or as valid to a particular situation as is desirable. The list of following articles is presented in chronological order of publication, with most recent first. A brief summary of each article is provided with some discussion of evaluation techniques used.

Intrusion Detection System Comparisons

The NSS Group. (2001). Intrusion Detection Systems Group Test (edition 2). [online] http://www.nss.co.uk

This is a comprehensive report on 15 commercial and o pen source Intrusion Detection Systems. The first edition of the report was published in 2000, using slightly different performance tests and evaluating a range of systems that were available at the time (some of these systems were also included in the 20 01 report). The NSS Group intend to continue to produce this report on a annual basis.

The evaluation of each IDS consists of two components. The first component is a qualitative analysis of the various features and functions of each product. This analysis is performed by IDS specialists, who have a range of experiences in the field. The comments and analysis of the various features are well considered and unbiased.

The quantitative component of consisted of four tests of the NIDSs on a controlled laboratory network. These test focused upon specific performance indicators, attack recognition, performance under load, ability to detect evasion techniques and a stateful operation test. The weakness of these tests is that the background traffic was generated using a Adtech AX/4000 broadband test system and a Smartbits SMB6000. Both of these traffic generators are designed to test network equipment, not Intrusion Detection Systems. Although the traffic generated consisted of valid IP packets, the traffic flow itself would be inconsistent with real life traffic. Two problems with this technique is firstly the likelihood of false positives being generated is reduced, if not eliminated and secondly that the actual attacks would differ significantly to t he background traffic.

The advantage of these traffic generators is that they are capable of generating sufficient traffic to saturate the network. However the relevance of this type of test on a IDS is debatable. In a production environment it is unlike ly that a network would be operating close to network saturation for any length of time. If this was the case the network would be redesigned or upgraded. For a in depth discussion of this topic see Ranum (2001)

The greatest critism of this testing proce ss is the lack of testing for false positive alerts. However this report is the most comprehensive, in terms of products tested and scientifically rigorous evaluation of Intrusion Detection Systems of which the author is aware. Any organisation contempla ting implementing a IDS must read this report.

Allen, J. Christie, A. William, F. McHugh, J. Pickel, J. Stoner, E. (2000) ***State of the Practice of Intrusion Detection Technologies.*** **Carnegie Mellon Software Engineering Institute.**

http://www.sei.cmu.edu/publications/documents/99.reports/99tr028/99tr028abstract.html

This publication covers a wide range if issues facing Intrusion Detection Issues, both in terms of functionality, performance and implementation. Section 3 of this paper discusses the performance of a number of IDSs available at the time. Although the authors did perform tests on various systems the testing methods and results are not directly mentioned. However this publication does discuss a wide range of issues relating to intrusion detection, and is highly critical of most of the systems tested at the time of publication. This document provides useful insights to important weaknesses of IDSs and a plethora of links to further information.

This publication also includes a list of recommended IDS selection criteria as a appendix. This list was originally published by Edwarrd Amoroso and Richard Kwapniewski. The author was unable to find a copy of the original document.

This list provides seven headings of topics of importance for IDSs. These are divided into two groupings, detection capabilities and operational capabilities.

Richard P. Lippmann, Robert K. Cunningham, David J. Fried, Issac Graf, Kris R. Kendall, Seth E. Webster, Marc A. Zissman(1999) ***. Results of the DARPA 1998 Offline Intrusion Detection Evaluation, slides presented at RAID 1999 Conference,*** **September 7-9, 1999, West Lafayette, Indiana.**

Haines, J, W. Lippmann, R, P. Fried, R, P. Korba, J. & Das, K. (1999) ***The 1999 DARPA Off-Line Intrusion Detection Evaluation.***

Haines, J, W. Lippmann, R, P. Fried, R, P. Zissman, M, A. Tran, E. & Bosswell, S, B. (1999) ***DARPA Intrusion Detection Evaluation: Design and Procedures. Lincoln Laboratory, Massachusetts Institute of Technology.***

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

This series of publications is a combined research effort from Lincoln Laboratory, DARPA and the American Air force. These combined publications refer to two comprehensive evaluations of IDSs and IDS technologies carried out on behalf of and with the assistance of DARPA. The aim of these evaluations were to assess the current state of IDS within the US defence and government organisations. These evaluations attempted to quantify specific performance measures of IDSs and test these against a background of realistic network traffic.

The performance measures used by these evaluation were: a ratio of attack detection to false positive, ability to detect new and stealthy attacks, a comparison of host vs. network based systems to detect different types of attacks, the ability of anomaly detection techniques to detect new attacks, improvements between 1998 and 1999, the ability of systems to accurately identify attacks. The research also attempted to establish the reason each IDS failed to detect a attack, or generated a false positive.

Both the 1998 and 1999 evaluations identified a number of weaknesses with existing IDSs. A number of these issues have since been resolved, while others are still valid. The testing process used sample of generated network traffic, audit logs, system logs and file system information. This information was then distributed to various evaluators who would provide the appropriate data to the Intrusion Detection

Systems. This ensured each system was provided with identical data, whilst allowing proper configuration of each system.

Other Evaluations

A number of mass media publications, both online and printed have published comparisons of Intrusion Detection Systems. However the articles reviewed by the author were lacking in scientific rigor and tended to depend upon qualitative evaluations, based solely upon the impression of the journalis t. The majority of these articles were extremely superficial in nature and in a some cases displayed a lack of understanding of IDS concepts by the relevant author. For this reason these articles have not been included.

IDS Evaluation Methodologies

Ranum, M, J. (2001). Experiences Benchmarking Intrusion Detection Systems. **NFR Security** http://www.nfr.com

This article discusses a number of issues relating to techniques used to benchmark (ie compare) IDSs. This article has the interesting perspective of a expert in the field and a vendor of a commercial IDS. This article is highly critical of many published IDS comparison for their lack of understanding of IDS techniques, and thus ability to design appropriate testing m ethodologies.

In particular Ranum discusses the various measures that can be and have been used measure the performance of IDSs. Recommended measures include a ratio of false positives to attacks and positives to attacks. The point is also made of the i mportance of using real life traffic and attacks in the evaluation process, rather than simulated traffic and attacks.

Alessandri, D. (2001). ***Using Rule-Based Activity Descriptions to Evaluate Intrusion Detection Systems.*** **:RAID 2001** http://www.raid-symposium.org/raid2001/program.html

Alessandri proposes the use of a systematic description scheme for regulating the descriptions used to describe IDS functions. This approach should allow for a evaluation of IDSs based upon their descriptions, without necessitating experimentation. The disadvantage of this approach is the requirement of accurate descriptions. Currently such a approach does not exist so implementing it is not possible. This approach does hold a certain promise for the future.

Puketza, N. Chung, M. Olsson, R, A. & Mukherjee, B. (1996). ***Simulating Concurrent Intrusions for Testing Intrusion Detection Systems: Parallelizing Intrusions.*** **University of California, Davis.**

Puketza, N. Zhang, K. Chung, M. Olsson, R, A. & Mukherjee, B. (1996). ***A Methodology for Testing Intrusion Detection Systems.*** **University of California, Davis.**

Puketza, N. Chung, M. Olsson, R, A. & Mukherjee, B. (1997) ***. A Software Platform for testing Intrusion Det ection Systems.*** **University of California, Davis.**

Due to the age of these documents the tests recommended are now quite dated. However the testing methodology used is still relevant. Puketza et al have developed a application to simulate specific attacks against a target system. These attacks can be scripted to run concurrently or in a specific sequence. The advantage of this

methodology is that each test can easily be repeated for each device under test. One disadvantage of this application is that it does target older vulnerabilities in UNIX systems, which should not apply to a current operating system. However this can easily be updated to include more contempary attacks.

Criteria for Evaluating Network Intrusion Detection Systems

The aim of this section is not to suggest a method of benchmarking NIDSs. Benchmarking as a method of evaluation is only valid in situations where the controlled environment has a close resemblance to the real life environment. As the performance of any NIDS is highly dependant upon its individual configuration, the network it is monitoring and it's position in that network benchmarking does not provide a definitive method of assessing a NIDS in a given situation. For further discussion on this topic see Ranum (2001). Rather this section aims to present a number of critera that can be used to determine the suitability of a given NIDS for a particular situation or environment.

The first step in the evaluation process should be to identify the importance of each of the topics listed in the following sections. The importance of individual criteria is likely to change from organisation to organisation. In many cases a topic will also require the identification of features specific to the network and systems to be monitored.

Measurement Criteria

Ability to identify attacks

The main performance requirement of a NIDS is to detect intrusions. However the definition of a intrusion is currently unclear. In particular, many vendors and researchers appear to consider any attempt to place malicious traffic on the network as a intrusion.

In reality a more useful system will log malicious traffic and only inform the operator if the traffic posses a serious threat to the security of the target host. Snort is tending towards this direction with the use a alert classification ranging from 1 to 10. With 1 representing a point of interest only and 10 representing a major threat to security.

Known vulnerabilities and attacks

All NIDSs should be capable of detecting known vulnerabilities. However research (Allen 2000), (NSS 2001) indicates that many commercial IDS fail to detect recently discovered attacks. On the other hand if a vulnerability or attack is known all systems should be patched, or workarounds applied thus the need for a NIDS to detect these events will be removed. Unfortunately the reality is that many systems are not patched or upgraded as vulnerabilities are discovered. This is clearly indicated by the number of system compromises that occur everyday, and the fact that most of the problems on the SANS top twenty list are predominantly old well known problems, with fixes available.

Unknown attacks

This must be the most important feature of any IDS. It is the IDS that can detect

attacks that are not yet known which will justify its implementation. New vulnerabilities are discovered every day. By its very nature these are also the most difficult attacks to detect.

Relevance of attacks

This refers to the ability of the NIDS to identify the relative importance of any attack. To return to the example already given the use of a windows exploit on a UNIX system is not of high importance. However if the alert is raised, and the analyst must investigate every alert, a mechanism should be available to distinguish the relevance of different alerts.

Stability, Reliability and Security

Any IDS should be able to continue consistantly operate in all circumstances. The application and operating system should be capable of running for years without segmentation faults or memory leakage.

A important function of a NIDS is to consistently report identical events in the same manner. One disadvantage of a product using signature recognition is the ability of different users to configure different alerts to provide different messages. Thus traffic on one network may trigger a different alert to the same traffic on another system of the same type. A number of efforts are currently underway to solve this problem. Both securityfocus and CVE provide databases of known vulnerabilities, and exploits targeting them.

The system should also be able to withstand attempts to compromise it. If a attacker can identify a NIDS on a network it will could prove to be a valuable asset. It is also possible the attacker will attempt to disable the system using DoS or DDoS techniques. The system should be able to withstand all of these types of attack.

Information provided to analyst

The information provided to the analyst when a alert is raised should be enough to clearly identify the reason the event causing the event to be raised, and the reason this event is of interest. It should also provide links to vulnerability databases, such as bugtraq or CVE to assist the analyst in determining the relevance and appropriate reaction to a particular alert.

Identify target and source

The alert should also identify the source of the alert and the target system. Further information such as a whois or DNS lookup on a IP address would be also be beneficial.

Severity, potential damage

Identification of the potential severity of a attack. Some alerts are triggered by events to related to information gathering, such as port scanning. Although this information may be relevant if a more serious attack in launched the volume of scanning that occurs on the internet makes it impractical to investigate every time a network is scanned. On the other hand indication that a local hosts has been compromised by a trojan should be given higher priority.

Outcome of attack (Success or failure)

Another useful (although currently non existent) feature of a NIDS should be to indicate the outcome of a attack. In most cases a alert simply indicates that a attempt has been made. It is then the responsibility of the analyst to search for correlating activity to indicate the outcome of the attack. If a NIDS were to present the analyst with a list of other alerts generated by the target host, and a summary of other (non alert) traffic the evaluation of the outcome could be greeted accelerated.

Legal validity of data collected

The legal validity of the data collected by any IDS is of extreme importance if any legal will be taken against the attacker. A disturbingly large number of systems do not collect the actual network packets, instead they simply record their own interpretation of events. A more robust systems will also capture and store the network traffic, as well as raising the alert.

Manageability

One of the greatest risks of a IDS is that once the system is implemented it will not be utilised to its full capabilities. Often the reason for this is due to the complexity of configuring and maintaining the system. It is also important that a IDS can be optimised for a particular network. There is no point in monitoring for web server exploits if there is not a web server on the network.

Ease or complexity of configuration

Unfortunately the usability of a system is usually inversely proportional to the flexibility and customisability of that system. The desire for flexibility can configurable of the system will be determined by the users of the system, the network in which it will be operating and the level of functionality required from the system.

If the system is to be maintained by a network administrator who is also responsible for standard network management he or she is unlikely to have the time available to optimise and configure the system so useability will be a primary consideration. On the other hand if a intrusion analyst if employed specifically to manage intrusion detection a more complex system with greater functionality may be desired.

Possible configuration options

The NIDS should be capable of being optimised for the systems on the network. As mentioned earlier there is no point in performing http analysis if a web server is not operating on the network under inspection. The level of traffic on the network will also determine the intensity of analysis performed. A simple system suitable for a single network segment with low traffic will be able to combine the sensor and analysis functions within the single unit. A network with high levels of traffic may need to separate the sensor and analysis functions across different hosts.

There are also a number of other configuration options that may apply to particular situations. For example in some situations the NIDS (i e analyst) may not be allowed to view the contents of packets on the network. In this case it should be possible to configure the NDIS to only examine (and store) the header information from the packets.

Scalability and interoperability

Scalability

Most organisations grow and expand over time. As they expand so do their supporting infrastructure, include computer networks. Any IDS should be capable of expanding with the network. As new network segments are added new NIDS may also be needed. Will it be possible to consolidate the reports from multiple NIDS into a single user interface? Another important question will be the storage of this information. If a small network is monitored data storage may be possible in flat files. However as the amount of data collected grows it may be necessary to transfer this data storage into a database.

Interoperability

Research has proven that the most effective intrusion detection requires correlating information from a range of sources. This includes NIDS, HIDS, system logs, firewall logs and any other information sources available. At the time of writing the Intrusion Detection Working Group (IDWG) had submitted a number of documents defining standards for communication between IDSs. It is expected that these will be released as RFCs in the near future.

Once these standards are implemented any IDS using the standard protocols will be able to communicate with and other IDS. This will enable a organisation to implement a range of IDS from different vendors and still maintain interoperability.

Vendor support

The level of vendor support required in a implementation will be determined by the skill levels of the staff implementing the system. However as staff turnover rates are common in the IT industry it is worthwhile considering the level of support that is available from the vendor.

Signature updates

Any signature based IDS is dependant upon it signatures to detect intrusions. The abilities of these systems to detect new, or even modified intrusions has been shown to be poor (Allen 2000). In order for these systems to be effective updated signatures must be available as new vulnerabilities and exploits are discovered.

Many signature based systems now allow the operator to create their own signatures. This can allow the system to monitor for new alerts as they are discovered without relying on the vendor to supply updates. However monitoring vulnerabilities and writing signatures as they occur is a demanding task. Consider the amount of traffic on bugtraq in a single day.

Conclusion

Selecting and implementing a NIDS is a challenging task. There are a number of factors to be considered, and these factors will change from situation to situation. In order to ensure a sucessful implementation a organisation should determine its requirements and then locate a system that meets them.

References:

The NSS Group. (2001). Intrusion Detection Systems Group Test (edition 2). [online] http://www.nss.co.uk

Allen, J. Christie, A. William, F. McHugh, J. Pickel, J. Stoner, E. (2000) *State of the Practice of Intrusion Detection Technologies.* Carnegie Mellon Software Engineering Institute. [online] http://www.sei.cmu.edu/publications/documents/99.reports/99tr028/99tr028abstract.html

Richard P. Lippmann, Robert K. Cunningham, David J. Fried, Issac Graf, Kris R. Kendall, Seth E. Webster, Marc A. Zissman(1999) . *Results of the DARPA 1998 Offline Intrusion Detection Evaluation, slides presented at RAID 1999 Conference,* September 7 -9, 1999, West Lafayette, Indiana. [on -line] http://www.ll.mit.edu/IST/ideval/index.html

Haines, J, W. Lippmann, R, P. Fried, R, P. Korba, J. & Das, K. (1999) *The 1999 DARPA Off - Line Intrusion Detection Evaluation.* [on-line] http://www.ll. mit.edu/IST/ideval/index.html

Haines, J, W. Lippmann, R, P. Fried, R, P. Zissman, M, A. Tran, E. & Bosswell , S, B. (1999) *DARPA Intrusion Detection Evaluation: Design and Procedures. Lincoln Laboratory, Massachusetts Institute of Technology.* [on-line] http://www.ll.mit.edu/IST/ideval/index.html

Ranum, M, J. (2001). *Experiences Benchmarking Intrusion Detection Systems* . NFR Security *http://www.nfr.com*

Ranum, M, J (1998) *Intrusion Detection: Challenges and Myths.* [On-Line] http://www.twinds.com/id -myths.html

Ranum, M, J (2001) *Coverage in Intrusion Detection Systems.* NFR Security [On-Line] http://www.nfr.com

Alessandri, D. (2001). Using Rule -Based Activity Descriptions to Evaluate Intrusion Detection Systems. *:RAID 2001 http://www.raid - symposium.org/raid2001/pro gram.html*

Puketza, N. Chung, M. Olsson, R, A. & Mukherjee, B. (1996). *Simulating Concurrent Intrusions for Testing Intrusion Detection Systems: Parallelizing Intrusions.* University of California, Davis.

Puketza, N. Zhang, K. Chung, M. Olsson, R, A. & Muk herjee, B. (1996). *A Methodology for Testing Intrusion Detection Systems.* University of California, Davis.

Puketza, N. Chung, M. Olsson, R, A. & Mukherjee, B. (1997). A Software Platform for testing Intrusion Detection Systems. *University of California, Davis.*

Amoroso, E. (1999). *Intrusion Detection.* USA. AT&T Laboratories

Maxion, R, A (1998). *Measuring Intrusion Detection Systems. Presented to the First Internation Workshop on Recent Advances in Intrusion Detection* 14-16 September, Louvain-la-Neuve, Belguim

Assignment 2 - Network Detects

The five detects listed here have been collected from different sources. A description of the detection and capture technique is provided with each capture

Trace 1 - Attempted exploit of Windows file sharing, by automated application.

Source of trace

This trace was made whilst using a dial up internet connection through a commercial ISP, from my laptop. I do not normally access the internet via a direct dial up connection, both at work and home I use Linux NAT and fir ewalling to protect my systems. On this occasion I was using a borrowed ISP account and dialing directly to the ISP, from a friends house. I had also forgotten that Samba had been installed on my system earlier during the day (very very poor security pra ctice).

Detect was generated by:

When using direct connections I routinely run ethereal to monitor the traffic to and from my system, especially when using a slow dial up connection as the level of traffic is fairly low. This trace occurred after 5 minu tes of the connection being made. Ethereal was running in full capture mode. This has made the detailed analysis of this attack possible.

Probability the source address was spoofed.

The source address was not spoofed. This trace shows a probe, using UDP and then a complete TCP connection, from SYN to FIN in response from the probe. If this address was spoofed the UDP packets would be from a different source.

Furthermore part of this trace includes responses from the attacker to information collected from previous packets. Although it would be theoretically possible to achieve this result using a packet sniffer and spoofed packets the probability of this is extremely low. Especially given the timeframe of the attack (10 seconds). It is likely this attack is a worm or automated scan, It is possible the owner of source system was not aware of its activity.

Description of Attack

Overview

This attack involves an attempt to connect to a shared folder called C on a microsoft windows 95, 98 or ME operating system. C is the default share name assigned when a user chooses to share their entire primary drive (or partition). This type of attempt is reasonably common on the internet. However in this case the point of interest is the speed with which the target system is identified and the attempt to use this shared drive is made, despite the fact the user has already gathered information indicating that there are no shared folders on the system. This leads to the conclusion that a automated tools is being used to scan for windows systems and immediately attempt to connect to shared folders. This may be a hacker utility or a worm.

A brief summary of the attack is listed below. A detailed interpretation (as provided by ethereal) of all packets is provided in Ap pendix A

No	*time*	*source*	*destination*	*Protocol*	*Information*
1	0.000000	161.184.237.101	210.50.17.34	NBNS	Name query NBSTAT *<00><0
2	0.000329	210.50.17.34	161.184.237.101	NBNS	Name query Response NBSTAT
3	6.459996	161.184.237.101	210.50.17.34	NBNS	Name query NBSTAT *<00><0
4	6.460253	210.50.17.34	161.184.237.101	NBNS	Name query Response NBSTAT
5	6.909988	161.184.237.101	210.50.17.34	TCP	1110 > 139 SYN
6	6.910050	210.50.17.34	161.184.237.101	TCP	139 > 1110 SYN ACK
7	7.319990	161.184.237.101	210.50.17.34	TCP	1110 > 139 ACK
8	7.320001	161.184.237.101	210.50.17.34	NBSS	Session request
9	7.320068	210.50.17.34	161.184.237.101	TCP	139 > 1110 ACK
10	7.903388	210.50.17.34	161.184.237.101	NBSS	Positive session response
11	8.419997	161.184.237.101	210.50.17.34	SMB	SMBnegprot Request
12	8.420049	161.184.237.101	210.50.17.34	TCP	139 > 1110 ACK
13	8.448496	210.50.17.34	161.184.237.101	SMB	SMBnegprot Response
14	8.889994	210.50.17.34	161.184.237.101	SMB	SMBsesssetupX Request
15	8.892277	161.184.237.101	210.50.17.34	SMB	SMBsesssetupX Response
16	9.379968	210.50.17.34	161.184.237.101	TCP	1110 >139 ACK
17	10.319995	210.50.17.34	161.184.237.101	TCP	1110 >139 FIN ACK
18	10.356431	161.184.237.101	210.50.17.34	TCP	139 > 1110 FIN ACK
19	10.759996	210.50.17.34	161.184.237.101	TCP	1110 > 139 ACK

Attack mechanism - Detailed Analysis

Step 1 - get netbios name

No	*time*	*source*	*destination*	*Protocol*	*Information*
1	0.000000	161.184.237.101	210.50.17.34	NBNS	Name query NBSTAT *<00><0
2	0.000329	210.50.17.34	161.184.237.101	NBNS	Name query Response NBSTAT
3	6.459996	161.184.237.101	210.50.17.34	NBNS	Name query NBSTAT *<00><0
4	6.460253	210.50.17.34	161.184.237.101	NBNS	Name query Response NBSTAT

The first packet from the attacker is a UDP netbios name query. The attacke r is attempting to find out the netbios name of my system, and the workgroup it belongs to. The * indicates he is looking for netbios information about any workgroup. The second packet is the response from my system. This gives the attacker my netbios name (MAPUCHE) and the workgroup the system belongs to, WORKGROUP.

Packets three and four are a repeat of this process. It is possible that the first response from my system was lost, or the software is use is faulty.

For a detail analysis of Netbios and S MB see http://samba.anu.edu.au/cifs/docs/what -is-smb.html

Step 2 - Establish a Netbios session

Now that the attacker knows the target systems name and workgroup he attempts to establish a SMB connection.

5	6.909988	161.184.237.101	210.50.17.34	TCP	1110 > 139 SYN
6	6.910050	210.50.17.34	161.184.237.101	TCP	139 > 1110 SYN ACK
7	7.319990	161.184.237.101	210.50.17.34	TCP	1110 > 139 ACK

5	6.909988	161.184.237.101	210.50.17.34	TCP	1110 > 139 SYN
8	7.320001	161.184.237.101	210.50.17.34	NBSS	Session request
9	7.320068	210.50.17.34	161.184.237.101	TCP	139 > 1110 ACK
10	7.903388	210.50.17.34	161.184.237.101	NBSS	Positive session response

Packets five to seven are a standard TCP handshake. As Samba is running on this system port 139 is open and a TCP se ssion is started.
Packet eight is a request to establish a Netbios session. The attacker supplies DAVE as his computer name. This is accepted by the target system which responds with packet ten. (Packet 9 is a plain ACK with no data)

Step 3 - Establish SMB protocol to be used

Now that a Netbios session has been established the attacker will try to establish a SMB session. Before this can happen the two systems must decide which SMB protocol to be used. This provides us with useful information about the operating system the attacker is using.

11	8.419997	161.184.237.101	210.50.17.34	SMB	SMBnegprot Request
12	8.420049	161.184.237.101	210.50.17.34	TCP	139 > 1110 ACK
13	8.448496	210.50.17.34	161.184.237.101	SMB	SMBnegprot Response

Packet eleven conta ins a request to negotiate SMB protocols, the attacker provides a list of protocols that he supports. The target then responds with packet thirteen, which indicates the protocol that will be used.

The decoded of the SMB contents of packet 11

```
SMB (Server Message Block Protocol)
  SMB Header
    Message Type: 0xFF
    Server Component: SMB
    SMB Command: SMBnegprot (0x72)
    Error Class: Success (0x00)
    Reserved: 0
    Error Code: No Error
    Flags: 0x00
      .... ...0 = Lock&Read, Write&Unlock not supported
      .... ..0. = Receive buffer not posted
      .... 0... = Path names case sensitive
      ...0 .... = Pathnames not canonicalized
      ..0. .... = OpLocks not requested/granted
      .0.. .... = Notify open only
      0... .... = Request to server
    Flags2: 0x0000
      .... .... .... ...0 = Long file names not supported
      .... .... .... ..0. = Extended attributes not supported
      .... .... .... .0.. = Security signatures not supported
      .... 0... .... .... = Extended security negotiation not supported
      ...0 .... .... .... = Don't resolve pathnames with DFS
      ..0. .... .... .... = Don't permit reads if execute-only
      .0.. .... .... .... = Error codes are DOS error codes
      0... .... .... .... = Strings are ASCII
    Reserved: 6 WORDS
    Network Path/Tree ID (TID): 0x0000
    Process ID (PID): 0x1b2f
    User ID (UID): 0x0000
    Multiplex ID (MID): 0x1881
  Word Count (WCT): 0
  Byte Count (BCC): 119
  Dialects
    Dialect Marker: 2
    Dialect: PC NETWORK PROGRAM 1.0
    Dialect Marker: 2
    Dialect: MICROSOFT NETWORKS 3.0
```

```
Dialect Marker: 2
Dialect: DOS LM1.2X002
Dialect Marker: 2
Dialect: DOS LANMAN2.1
Dialect Marker: 2
Dialect: Windows for Workgroups 3.1a
Dialect Marker: 2
Dialect: NT LM 0.12
```

One point of interest with this response is the fact that the flag to accept long filenames is not set. This would be expected for DOS system, but not for a windows system.

The dialects support is of interest to us as it provides some indication of the attacking operating system. In this case as NT LM 0.12 is offered it is likely that attacking OS is Windows NT or 2000.

The target system responds with packet 13.

```
SMB (Server Message Block Protocol)
  SMB Header
    Message Type: 0xFF
    Server Component: SMB
    SMB Command: SMBnegprot (0x72)
    Error Class: Success (0x00)
    Reserved: 0
    Error Code: No Error
    Flags: 0x80
      .... ...0 = Lock&Read, Write&Unlock not supported
      .... ..0. = Receive buffer not posted
      .... 0... = Path names case sensitive
      ...0 .... = Pathnames not canonicalized
      ..0. .... = OpLocks not requested/granted
      .0.. .... = Notify open only
      1... .... = Response to client/redirector
    Flags2: 0x0001
      .... ...1 .... ...1 = Long file names supported
      .... .... .... ..0. = Extended attributes not supported
      .... .... .... .0.. = Security signatures not supported
      .... 0... .... .... = Extended security negotiation not supported
      ...0 .... .... .... = Don't resolve pathnames with DFS
      ..0. .... .... .... = Don't permit reads if execute-only
      .0.. .... .... .... = Error codes are DOS error codes
      0... .... .... .... = Strings are ASCII
    Reserved: 6 WORDS
    Network Path/Tree ID (TID): 0x0000
    Process ID (PID): 0x1b2f
    User ID (UID): 0x0000
    Multiplex ID (MID): 0x1881
  Word Count (WCT): 17
  Dialect Index: 5, Greater than LANMAN2.1
  Security Mode: 0x03
    .... ...1 = Security = User
    .... ..1. = Passwords = Encrypted
    .... .0.. = Security signatures not enabled
    .... 0... = Security signatures not required
  Max multiplex count: 50
  Max vcs: 1
  Max buffer size: 65535
  Max raw size: 65536
  Session key: 000004cb
  Capabilities: 0x03b9
    .... .... .... .... .... .... ...1 = Raw Mode supported
    .... .... .... .... .... .... ..0. = MPX Mode not supported
    .... .... .... .... .... .... .0.. = Unicode not supported
    .... .... .... .... .... .... 1... = Large files supported
    .... .... .... .... .... ...1 .... = NT LM 0.12 SMBs supported
    .... .... .... .... .... ..1. .... = RPC remote APIs supported
    .... .... .... .... .... .0.. .... = NT status codes not supported
    .... .... .... .... .... 1... .... = Level 2 OpLocks supported
    .... .... .... .... ...1 .... .... = Lock&Read supported
    .... .... .... .... ..1. .... .... = NT Find supported
```

```
      .... .... .... .... ...0 .... .... .... = DFS not supported
      .... .... .... .... .0.. .... .... .... = Large READX not supported
      .... .... .... .... 0... .... .... .... = Large WRITEX not supported
      0... .... .... .... ...- .... .... .... = Extended security exchanges not supported
    System Time Low: 0x8c04eb80
    System Time High: 0x01c1c723
    Server time zone: -660 min from UTC
    Encryption key len: 8
    Byte count (BCC): 18
    Challenge encryption key: 41CB66C290722F42
    OEM domain name: WORKGROUP
```

At this point we can see that the target system will support NT LM0.12 and accepts long filenames. It also informs the attacker that th e domain name is WORKGROUP. This is the default name given when Microsoft networking in installed, either with SAMBA or on microsoft operating systems. It would be a sensible NT domain name for a attacker running a scan to use.

The attacker has also be en informed that user level security has been applied, and that encrypted passwords are expected.

Step 4 - Try and access a shared drive

14	8.889994	210.50.17.34	161.184.237.101	SMB	SMBsesssetupX Request
15	8.892277	161.184.237.101	210.50.17.34	SMB	SMBsesssetupX Response

Packet 14 is the request to establish a SMB session and access the shared C drive. This packet also provides us with a large amount of information about the attacking system. We know that the users name is Dave McGrath and he has conf igured his domain name as HOME. He has also supplied a encrypted password. It would be interesting to use this password back against his own system and see what we would find.

```
SMB (Server Message Block Protocol)
  SMB Header
    Message Type: 0xFF
    Server Component: SMB
    SMB Command: SMBsesssetupX (0x73)
    Error Class: Success (0x00)
    Reserved: 0
    Error Code: No Error
    Flags: 0x10
      .... ...0 = Lock&Read, Write&Unlock not supported
      .... ..0. = Receive buffer not posted
      .... 0... = Path names case sensitive
      ...1 .... = Pathnames canonicalized
      ..0. .... = OpLocks not requested/granted
      .0.. .... = Notify open only
      0... .... = Request to server
    Flags2: 0x0000
      .... ...- .... ...0 = Long file names not supported
      .... ...- .... ..0. = Extended attributes not supported
      .... ...- .... .0.. = Security signatures not supported
      .... 0... .... .... = Extended security negotiation not supported
      ...0 .... .... .... = Don't resolve pathnames with DFS
      ..0. .... .... .... = Don't permit reads if execute-only
      .0.. .... .... .... = Error codes are DOS error codes
      0... .... .... .... = Strings are ASCII
    Reserved: 6 WORDS
    Network Path/Tree ID (TID): 0x0000
    Process ID (PID): 0x1b2f
    User ID (UID): 0x0001
    Multiplex ID (MID): 0x1881
  Word Count (WCT): 13
  AndXCommand: SMBtconX
  AndXReserved: 0
```

```
AndXOffset: 127
MaxBufferSize: 2920
MaxMpxCount: 50
VcNumber: 0
SessionKey: 1227
ANSI Account Password Length: 24
UNICODE Account Password Length: 0
Reserved: 0
Capabilities: 0x00000001
   .... .... .... .... .... .... .... ...1 = Raw Mode supported
   .... .... .... .... .... .... .... ..0. = MPX Mode not supported
   .... .... .... .... .... .... .... .0.. = Unicode not supported
   .... .... .... .... .... .... .... 0... = Large Files not supported
   .... .... .... .... .... .... ...0 .... = NT LM 0.12 SMBs not supported
   .... .... .... .... .... .... ..0. .... = RPC Remote APIs not supported
   .... .... .... .... .... .... .0.. .... = NT Status Codes not supported
   .... .... .... .... .... .... 0... .... = Level 2 OpLocks not supported
   .... .... .... .... .... ...0 .... .... = Lock&Read not supported
   .... .... .... .... .... ..0. .... .... = NT Find not supported
   .... .... .... .... ...0 .... .... .... = DFS not supported
   .... .... .... .... .0.. .... .... .... = Large READX not supported
   .... .... .... .... 0... .... .... .... = Large WRITEX not supported
   0... .... .... .... .... .... .... .... = Extended Security Exchanges not supported
Byte Count: 66
ANSI Password: \2135?\033\341\362gM\273z\0325n\362%\346\200\004\037~w\376,g
Account Name: DAVE MCGRATH
Primary Domain: HOME
Native OS: Windows 4.0
Native LanMan Type: Windows 4.0
Word Count (WCT): 4
Next Command: No further commands
Reserved (MBZ): 0
Offset to next command: 0
Additional Flags: 0x0002
   .... .... .... ...0 = Don't disconnect TID
Password Length: 1
Byte Count (BCC): 16
Password: \000
Path: \\MAPUCHE\C
Service: A:
```

Packet 15 is the response from the target system, informing the attacker that a bad password has been supplied. It is assumed that the attacker was only looking for systems with shared folders unprotected by passwords. As no further attempt is made to target this system, or try multiple passwords.

Correlations

This attack is reasonably common and simple. The area of interest here is not the attack itself, but the fact it has been automated. The user is obviously using a scanning tool to find windows 95, 98 or ME systems with file sharing enabled and the C drive shared. Scanners to perform this task are not uncommon, for example winhackgold, smbscanner, smbls98 and msmbs are all designed with this goal in mind. However none of these applications attempt to scan and connect to a shared 'C' folder simultaneously. I have been unable to find a reference to this particular type of scan. This may be a new tool, or a variant of a old one.

A whois reveals that the source address is owned by Edmonton Telephones Corporation. The contact address of telusplanet.net appears to be a ISP. It is probable this connection is from a ISP account.

> Edmonton Telephones Corporation (NET-ED-TEL)
> 355 - 4th Ave SW Suite 300 Calgary, AB
> T2P 0J1 CA Netname: ED-TEL
>
> Netblock: 161.184.0.0 - 161.184.255.255

Maintainer: EDTC Coordinator: TELUS Communications (FTS1 -ARIN)
hostmaster@telusplanet.net +1 877 310 -4638
Domain System inverse mapping provided by:
CLGRP S01.AGT.NET 198.80.55.1
CLGRPS02.AGT.NET 198.161.156.1
Record last updated on 12 -Feb-2001.
Database last updated on 9 -Mar-2002 19:56:49 EDT.
The ARIN Registration Services Host contains ONLY Internet Network Information: Networks, ASN's, and related PO C's. Please use the whois server at rs.internic.net for DOMAIN related Information and whois.nic.mil for NIPRNET Information.

A reverse DNS lookup reveals a DNS name of edtn015849.hs.telusplanet.net .

A report of activity on port 139 on dshield reveals that activity on port 139 has ranged between 0.2% and 1.6% of activity over most of March. This is a fairly high level for a single port.

Evidence of active targetin g

There is little indication of targeting the specific host mentioned in this attack. However given the speed of the attack it is likely a scan of the subnet was in progress. The target system was using a ISP connection, commonly used by home users, who are likely to have configured unprotected SMB shared folders. So there is a high probability that the target was the ISP subnet, rather than the particular host. The facts that once the attack failed and no more attempts were made to target the host were made and that the host had only been using the target IP address for 5 minutes supports this argument.

Severity

Criticality – This attack targets system using the microsoft windows 95, 98 and ME operating systems. These operating systems do not provide significant levels of security and would not be used to support significant internet functions. Thus this attack targets desktop systems not servers. criticality = 1

Lethality – If this attack was successful the attacker would have complete control over the target system. Trojans could be copied onto the target and operating system files modified to run the trojans at system boot. Lethality = 5

System Countermeasures – In this case the target system was using Linux, and Samba, which is not susceptib le to this type of attack, unless specifically configured that way. The microsoft operating systems targeted by this attack can implement passwords on shared files, this would prevent this specific attack. System Countermeasures = 4

Network Countermeasu res – The network administrators should block port 135, 137 and 139 . This would defend against this attack. This is standard in most corporate environments. The ISP in this case could also implement this countermeasure without affecting the majority of their clients. Network Countermeasures = 5

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

(1 + 5) – (4 + 5) = -3

Defensive Recommendation:

The system targeted by this attack was configured be impervious to this attack. This type of attack could also be defended against by the ISP routers. See network countermeasures for details.

Multiple choice test question:

Which TCP/IP protocol is the exception to the rule that clients only used emphereal ports?

a) http
b) FTP
c) Netbios
d) SMTP
e) ARP

Correct answer **c**

Trace 2 - Netlogon response to private address

Source of Trace

This trace was made on our local staff network. This nearly a 100% windows network. There are two windows Domains, staff and student. The student networ k is behind a commercial NAT system, using the private network 192.168.0.0/24. The staff network is using real IP addresses, with some static address and the majority dynamic addresses using DHCP. The staff network actually uses two class C networks. The initial detect was made by a sniffer on the segment of the staff network containing the staff Backup Domain Controller (BDC), fileservers, WINS, DNS and DHCP server.

Detect was generated by:

The original detect was generated by a prototype NIDS currentl y under development. This system uses a combination of anomaly and profiling detection techniques. This uses a packet capture engine to store packets for later analysis.

The follow up detects and further investigation were made using TCPdump on the local network with filters to capture the specific activity of interest.

Probability the source address was spoofed

At first appearances the probability the source address was spoofed appeared fairly high. However as shall be shown in the analysis the source address was not spoofed.

Description of attack

This trace does not relate directly to a attack. The original detect is of a netlogon response from the BDC to a private address 192.168.0.56. This would normally be a response to a netlogon request. No oth er packets were identified with either a source or destination of 192.168.0.56. As the response address was a private address, in a public network this was not surprising. However it appeared that the BDC was being stimulated to send this response.

The original packet that triggered that alert was simply a UDP packet with a source of MY.NET.ONE.15:138 and destination of 192.168.0.56:138.

The source and destination ports of 138 indicate NetBIOS packets.

These alerts occurred on a number of occasions with a range of different 192.168.0.0/24 destination addresses.

At this point tcpdump was run on the network with a filter to collect all packets with a source or destination address in 192.168.0.0/24 range. A number of these packets were collected. A etherea l decode of the netbios section of this packet is shown below.

```
Frame 19 (239 on wire, 239 captured)
NetBIOS Datagram Service
   Message Type: Direct_unique datagram (16)
   More fragments follow: No
   This is first fragment: Yes
   Node Type: NBDD (3)
   Datagram ID: 0xc66d
   Source IP: MY.NET.ONE.15 (MY.NET.ONE.15)
   Source Port: 138
```

Datagram length: 183 bytes
Packet offset: 0 bytes
Source name: STAFF -BDC<00> (Workstation/Redirector)
Destination name: LAB18 -13 <00> (Workst ation/Redirector)

Two significant pieces of information are displayed here. Firstly the destination netbios name is LAB18 -13. This indicates machine number 13 in computer lab 1.8. We now know the source of the request. Secondly the netbios information also includes a source IP address. This lead to the conclusion that a student was able to send a network logon request through the NAT firewall to the staff BDC. As the netbios information included with the logon request also included the source address , this would not have been changed by the NAT. It appears that the BDC was in fact responding to the netbios source address rather than the source IP address. This was confirmed by capturing more packets.

The final packet shown here is a net logon reques t from the NAT to the BDC on behalf of the student machine.

Internet Protocol, Src Addr: MY.NET.ONE.23 **(MY.NET.ONE.23)** , Dst Addr: MY.NET.ONE.15 (MY.NET.ONE.15)
User Datagram Protocol, Src Port: 138 (138), Dst Port: 138 (138)
Source port: 138 (138)
Destination port: 138 (138)
Length: 228
Checksum: 0x03bf (correct)
NetBIOS Datagram Service
Message Type: Direct_group datagram (17)
More fragments follow: No
This is first fragment: Yes
Node Type: B node (0)
Datagram ID: 0 x0010
Source IP: 192.168.0.128 (192.168.0.128)
Source Port: 138
Datagram length: 206 bytes
Packet offset: 0 bytes
Source name: LAB18 -18 <00> (Workstation/Redirector)
Destination name: STAFF -BDC <1c> (Domain Controllers)

In this final packet it can be seen how the netbios source address 192.168.0.128 (the lab machine) is different to the packet source address of MY.NET.ONE.23 (the NAT router).

Attack mechanism

This attack consisted of a machine being configured to logon to the Staff domain, rather than the student domain. This is a simple option available at the windows logon prompt.

In order for this attack to succeed the student would have to know a valid username and password for the domain. This was not the case as th e username supplied was in fact the students standard logon.

Microsoft Windows Logon Protocol
Command: LM1.0/LM2.0 LOGON Request (0x00)
Computer Name: LAB18 -18
User Name: STUDENTS_ID
Mailslot Name: \MAILSLOT \TEMP \NETLOGON
Request Coun t: 0
NT Version: 2

LM20 Token: 0xffff (LanMan 2.0 or higher)

This attack would have been simple to execute as both the staff and student networks use the same WINS server. The student could have queried this server to find the Staff BDC. As the operating systems used in the computer labs is Windows 98 there is little control over a users actions on the lab machines.

Correlations

Correlations to this attack could have been found in the NT log files on the BDC. Unfortunately I do not have access t o these, and I am under the impression that failed logon attempts are not recorded anyway.

Evidence of active targeting

The evidence from the actual packets captured indicate a high level of active targeting. The attacking host was specifically targeting the BDC. However the student attempting this attack was simply trying to access the staff domain, not the BDC. The target of this attack was the STAFF domain, rather than a single host.

Severity

Criticality – The target of this attack was a windows domai n. This domain does contain confidential information. Criticality = 3

Lethality – If the attacker was able to successfully gain access to the STAFF domain potentially could have corrupted staff files. Lethality = 2

System Countermeasures – In order for this attack to succeed a valid username and password were needed. System Countermeasures = 4

Network Countermeasures – The NAT router was configured to block TCP connections to the BDC, it has also now been configured to block UDP packets. This attack is no longer possible from the student network. Network Countermeasures = 5

(3 + 2) – (4 + 5) = -4

Defensive recommendations

All users should use good passwords. The NAT router should (and now is) configured to block all connections to staff machines.

Multiple choice test question

When using Network Address Translation the local hosts:

a) are hidden from the external network

b) are only accessable to external who know their local IP address

c) can connect to any external host, only if the external host initialises the connection

d) are still exposed to attackers on the external network

Correct answer **a**

Trace 3 - IIS Vulnerability scan

Source of Trace

This trace was collected from a network segment containing three web servers. These servers are used to h ost the sub faculty website, student WebPages and a commercial online learning shell. The server hosting the Sub Faculty website has a valid DNS entry and is linked to from the main website. It contains information that may be of interest to the general public. The other two servers are used for teaching purposes only. The only reason for these systems to be accessed from outside the university would be students or lecturers working from home. The student web server is only used in two subjects for tea ching website design and does not even have a DNS entry.

Detect was generated by:

These detects were generated by Snort 1.8.3 using a 1.8.1 rule set downloaded on 15/03/2002. For the purpose of this assignment tcpdump was also running in full capture mode on the same network. (both of these systems were using network taps).

Probability the source address was spoofed

The source addresses in this detect were not spoofed. The range of attacks used all relied on a TCP connection and expected responses from their stimuli.

Description of attack

This attack consisted of attempts to exploit 33 different vulnerabilities with IIS servers.

This attack was performed with a vulnerability scanner, designed to test for known vulnerabilities in microsoft Internet Infor mation Server, and targeted all three web servers. This attack was used on numerous occasions by a number of different hosts, up to 6 times a day. This is a excellent example of the importance of maintaining patches and upgrades on systems exposed to the internet.

Attack mechanism

The actual attack mechanism is a automated vulnerability scanner. The list of Snort alerts, rules and a brief description of each alert, generated by this scan are listed below

```
[**] [1:1256:1] WEB-IIS CodeRed v2 root.exe access [**]
[Classification: Attempted Administrator Privilege Gain] [Priority: 10]
03/20-23:41:14.765925 203.235.201.197:3885 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:6756 IpLen:20 DgmLen:112 DF
***AP*** Seq: 0xEDA6DBD5  Ack: 0x2633F96  Win: 0x2238  TcpLen: 20

[**] [1:1256:1] WEB-IIS CodeRed v2 root.exe access [**]
[Classification: Attempted Administrator Privilege Gain] [Priority: 10]
03/20-23:41:14.783477 203.235.201.197:3885 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:112
***AP*** Seq: 0xD5DBA6ED  Ack: 0xD5DBA6ED  Win: 0x21F0  TcpLen: 20
```

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB -IIS CodeRed v2 root.exe access"; flags: A+; uricontent:"scripts/root.exe?"; nocase; classtype: attempted -admin; sid: 1256; rev: 1;)

The first two alerts have both triggered the same rule. This rule is triggered by any

attempt to access scripts/root.exe. This reason this alert is identified as Code Red v2 root.exe access is that Code Red V2 places a executable file root.exe in the web servers script directory. However if this was a Code Red 2 worm it would not attempt to connect twice.

There are three interesting difference between the two alerts. The first detected packet has a TTL of 110, a TOS of 0x0 and the do not fragment flag set.

The second code red 2 packet arrives .02 seconds after the first packet. This packet is using the same tcp port (3885), yet the sequence number is lower, the TTL is 255 (this is worrying as it may indicate that the attacker is on the local network), the ID is 0, the TOS is 0x10 and the sequence number and ack are identical. It is possible that the packet to arrive second was in fact the first packet sent, however it still should not have a ID of 0 or a TTL of 255. There is a high probability that this pac ket has been crafted.

According to RFC 791if bit 3 in the TOS field is set to 1 a low delay service is requested. This should not affect the manner by which the TTL is decremented, and may or may not be acknowledged by particular routers.

The final point of interest is that the length of both packets is identical (112 octets). This introduces the possibility that both packets are carrying the same payload.

It is most likely this alert relates to a attempt by the attacker to identify if this system has been infected by the Code Red 2 worm, which has left behind a backdoor.

http://www.eeye.com/html/Research/Advisories/AL20010804.html

```
[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:16.060456 203.235.201.197:3900 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:21605 IpLen:20 DgmLen:120 DF
***AP*** Seq: 0xEDA6DC96  Ack: 0x2633FB7  Win: 0x2238  TcpLen: 20

[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:16.078828 203.235.201.197:3900 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:120
***AP*** Seq: 0x96DCA6ED  Ack: 0x96DCA6ED  Win: 0x21E8  TcpLen: 20

[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:16.705507 203.235.201.197:3908 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:59237 IpLen:20 DgmLen:120 DF
***AP*** Seq: 0xEDA6DCE3  Ack: 0x2633FC8  Win: 0x2238  TcpLen: 20

[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:16.723247 203.235.201.197:3908 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:120
***AP*** Seq: 0xE3DCA6ED  Ack: 0xE3DCA6ED  Win: 0x21E8  TcpLen: 20
```

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB -IIS cmd.exe access"; flags: A+; content:"cmd.exe"; nocase; classtype:attempted -user; sid:1002; rev:1;)

The next four alerts are all triggered by any packet containing the string "cmd.exe". This is most likely a attempt to execute a command on the target system.

The unusual pattern of the second packet having a ID of 0, TTL of 255, TOS= 0x10, each packet of equal length and iden tical sequence and acknowledgment numbers is repeated again for both of these pairs of alerts. The first packet also has the DF flag

set, as was the case for the Code Red alerts.

At this point the increase in port numbers should also be noted. The attack er has jumped from port 3885 (in the code red alert) to 3900 in 1.3 seconds. This high jump in ports implies that a scanner is in use and other targets are processed after our host has been probed.

```
[**] [1:974:2] WEB-IIS ..\.. access [**]
[Classification: Attempted Information Leak] [Priority: 3]
03/20-23:41:17.360558 203.235.201.197:3917 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:33894 IpLen:20 DgmLen:136 DF
***AP*** Seq: 0xEDA6DD54  Ack: 0x2633FD7  Win: 0x2238  TcpLen: 20
[Xref=> http://www.securityfocus.com/bid/2218]
[Xref=> http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0229]

[**] [1:974:2] WEB-IIS ..\.. access [**]
[Classification: Attempted Information Leak] [Priority: 3]
03/20-23:41:17.379134 203.235.201.197:3917 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:136
***AP*** Seq: 0x54DDA6ED  Ack: 0x54DDA6ED  Win: 0x21D8  TcpLen: 20
[Xref=> http://www.securityfocus.com/bid/2218]
[Xref=> http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0229]
```

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB -IIS ..\.. access";flags: A+; content:"|2e2e5c2e2e|"; reference:bugtraq,2218; reference:cve,CAN-1999-0229; classtype:attempted-recon; sid:974; rev:2;)

This alert is designed to detect attempt to access files outside the website root directory. This exploit is relatively old and only affects IIS 1.0 and possibly IIS 2.0. It is unusual to see this old exploit here as the probability of finding a server vulnerable to it is extremely low.

However there are a number of other exploits that might match this rule. The length of the packet is 136 octets.

Again the unusual features of the second packet can be observed.

```
[**] [1:1288:1] WEB-FRONTPAGE /_vti_bin/ access [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
03/20-23:41:18.005456 203.235.201.197:3923 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:9575 IpLen:20 DgmLen:157 DF
***AP*** Seq: 0xEDA6DD8C  Ack: 0x2633FE7  Win: 0x2238  TcpLen: 20

[**] [1:1288:1] WEB-FRONTPAGE /_vti_bin/ access [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
03/20-23:41:18.024818 203.235.201.197:3923 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:157
***AP*** Seq: 0x8CDDA6ED  Ack: 0x8CDDA6ED  Win: 0x21C3  TcpLen: 20
```

alert tcp $EXTERNAL_NET any -> $HTTP_SERVERS 80 (msg:"WEB -FRONTPAGE /_vti_bin/ access";flags: A+; uricontent:"/_vti_bin/"; nocase; classtype:web-application-activity; sid:1288; rev:2;)
This alert relates to attempts to access the _vti_bin directory. This normally contains microsoft frontpage extensions, which provide functionality for dynamic web pages. As frontpage extensions are not installed on the target system this is not a threat. Note that port numbers are still rising

```
[**] [1:974:2] WEB-IIS ..\.. access [**]
[Classification: Attempted Information Leak] [Priority: 3]
03/20-23:41:18.644451 203.235.201.197:3935 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:51559 IpLen:20 DgmLen:157 DF
***AP*** Seq: 0xEDA6DE0A  Ack: 0x2634006  Win: 0x2238  TcpLen: 20
[Xref=> http://www.securityfocus.com/bid/2218]
```

```
[Xref=> http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0229]

[**] [1:974:2] WEB-IIS ..\.. access [**]
[Classification: Attempted Information Leak] [Priority: 3]
03/20-23:41:18.662226 203.235.201.197:3935 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:157
***AP*** Seq: 0xADEA6ED  Ack: 0xADEA6ED  Win: 0x21C3  TcpLen: 20
[Xref=> http://www.securityfocus.com/bid/2218]
[Xref=> http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0229]

[**] [1:974:2] WEB-IIS ..\.. access [**]
[Classification: Attempted Information Leak] [Priority: 3]
03/20-23:41:19.297256 203.235.201.197:3941 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:24936 IpLen:20 DgmLen:185 DF
***AP*** Seq: 0xEDA6DE57  Ack: 0x2634019  Win: 0x2238  TcpLen: 20
[Xref=> http://www.securityfocus.com/bid/2218]
[Xref=> http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0229]

[**] [1:974:2] WEB-IIS ..\.. access [**]
[Classification: Attempted Information Leak] [Priority: 3]
03/20-23:41:19.315462 203.235.201.197:3941 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:185
***AP*** Seq: 0x57DEA6ED  Ack: 0x57DEA6ED  Win: 0x21A7  TcpLen: 20
[Xref=> http://www.securityfocus.com/bid/2218]
[Xref=> http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0229]
```

We now see another four packets that have triggered the .. \.. rule. However this time the length of the packets has increased from 136 octets to 157 in the second packet and 185 in the final packet. It is likely attempts are being made to use different exploits.

```
[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:19.932451 203.235.201.197:3949 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:62312 IpLen:20 DgmLen:137 DF
***AP*** Seq: 0xEDA6DE97  Ack: 0x263402E  Win: 0x2238  TcpLen: 20

[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:19.950456 203.235.201.197:3949 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:137
***AP*** Seq: 0x97DEA6ED  Ack: 0x97DEA6ED  Win: 0x21D7  TcpLen: 20

[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:20.575499 203.235.201.197:3956 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:33897 IpLen:20 DgmLen:137 DF
***AP*** Seq: 0xEDA6DEE3  Ack: 0x2634035  Win: 0x2238  TcpLen: 20

[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:20.591528 203.235.201.197:3956 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:137
***AP*** Seq: 0xE3DEA6ED  Ack: 0xE3DEA6ED  Win: 0x21D7  TcpLen: 20

[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:21.208921 203.235.201.197:3964 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:5226 IpLen:20 DgmLen:137 DF
***AP*** Seq: 0xEDA6DF2B  Ack: 0x2634039  Win: 0x2238  TcpLen: 20

[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:21.231109 203.235.201.197:3964 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:137
***AP*** Seq: 0x2BDFA6ED  Ack: 0x2BDFA6ED  Win: 0x21D7  TcpLen: 20

[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:21.869047 203.235.201.197:3973 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:47722 IpLen:20 DgmLen:137 DF
***AP*** Seq: 0xEDA6DF74  Ack: 0x263403D  Win: 0x2238  TcpLen: 20
```

[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:21.887264 203.235.201.197:3973 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:137
AP Seq: 0x74DFA6ED Ack: 0x74DFA6ED Win: 0x21D7 TcpLen: 20

This is a repeat of packets containing the phrase "cmd.exe". Each of these packets is of the same length, however source ports are still incre asing.

[**] [1:974:2] WEB-IIS ..\.. access [**]
[Classification: Attempted Information Leak] [Priority: 3]
03/20-23:41:22.510828 203.235.201.197:3982 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:24939 IpLen:20 DgmLen:138 DF
AP Seq: 0xEDA6DFCC Ack: 0x2634041 Win: 0x2238 TcpLen: 20
[Xref=> http://www.securityfocus.com/bid/2218]
[Xref=> http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0229]

[**] [1:974:2] WEB-IIS ..\.. access [**]
[Classification: Attempted Information Leak] [Priority: 3]
03/20-23:41:22.535770 203.235.201.197:3982 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:138
AP Seq: 0xCCDFA6ED Ack: 0xCCDFA6ED Win: 0x21D6 TcpLen: 20
[Xref=> http://www.securityfocus.com/bid/2218]
[Xref=> http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0229]

[**] [1:974:2] WEB-IIS ..\.. access [**]
[Classification: Attempted Information Leak] [Priority: 3]
03/20-23:41:23.163945 203.235.201.197:3989 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:3948 IpLen:20 DgmLen:136 DF
AP Seq: 0xEDA8D41C Ack: 0x2634043 Win: 0x2238 TcpLen: 20
[Xref=> http://www.securityfocus.com/bid/2218]
[Xref=> http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0229]

[**] [1:974:2] WEB-IIS ..\.. access [**]
[Classification: Attempted Information Leak] [Priority: 3]
03/20-23:41:23.180789 203.235.201.197:3989 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:136
AP Seq: 0x1CD4A8ED Ack: 0x1CD4A8ED Win: 0x21D8 TcpLen: 20
[Xref=> http://www.securityfocus.com/bid/2218]
[Xref=> http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0229]

[**] [1:974:2] WEB-IIS ..\.. access [**]
[Classification: Attempted Information Leak] [Priority: 3]
03/20-23:41:23.830627 203.235.201.197:3999 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:47724 IpLen:20 DgmLen:140 DF
AP Seq: 0xEDA8D477 Ack: 0x263404B Win: 0x2238 TcpLen: 20
[Xref=> http://www.securityfocus.com/bid/2218]
[Xref=> http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0229]

[**] [1:974:2] WEB-IIS ..\.. access [**]
[Classification: Attempted Information Leak] [Priority: 3]
03/20-23:41:23.849122 203.235.201.197:3999 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:140
AP Seq: 0x77D4A8ED Ack: 0x77D4A8ED Win: 0x21D4 TcpLen: 20
[Xref=> http://www.securityfocus.com/bid/2218]
[Xref=> http://cve.mitre.org/cgi-bin/cvename.cgi?name=CAN-1999-0229]

Again we see attempts to use .. \... The length of packets is still changing as is the source port.

[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:24.479445 203.235.201.197:4005 -> MY.NET.ONE.18:80
TCP TTL:110 TOS:0x0 ID:24685 IpLen:20 DgmLen:136 DF
AP Seq: 0xEDA8D4B7 Ack: 0x2634052 Win: 0x2238 TcpLen: 20

[**] [1:1002:1] WEB-IIS cmd.exe access [**]
[Classification: Attempted User Privilege Gain] [Priority: 8]
03/20-23:41:24.497065 203.235.201.197:4005 -> MY.NET.ONE.18:80
TCP TTL:255 TOS:0x10 ID:0 IpLen:20 DgmLen:136
AP Seq: 0xB7D4A8ED Ack: 0xB7D4A8ED Win: 0x21D8 TcpLen: 20

Finally the scan finishes with a last attempt to use cmd.exe. Again the source port has increased.

Analysis

The aim of this scan has been to test number of known vulnerabilities with Inter net Information Server. From a analysis perspective the most intriguing information is the apparent repeat of packets with a TTL of 255 and identical sequence and acknowledgment numbers. Further investigation of the snort log files reveals that a number of other unrelated alerts are also displaying this sort of behaviour. Given that a packet with a TTL of 255 could only reach target from the local network, the fact that the alerts all relate to TCP sessions and considering the unusual sequence numbers and IP ID numbers (of 0) it is impossible these packets were part of a genuine TCP session. It is most likely these second packets are the result of a misconfiguration of Snort or a network device. The attacker could not benefit from placing these packets on the network.

The consistent increase in source port number indicates that the attacker is opening other ports in between connecting to the target system. The increment in port number is between 6 and 9 on all but two occasions. This sort of pattern wo uld be expected if a scan of other hosts is being carried out simultaneously. The short time frame between alerts (less than 1 second between each alert) increases the probability of a scan taking place.

This scanning tool is not particularly well coded. The tool attempts minor variations of an exploits dependant upon the same vulnerability after the first attempt has already failed. Indicates a fairly simplistic application. The scanner also does not appear to made any effort to elude detection.

The fact that a identical pattern of alerts is generated six times on the same day this alert was generated, from a range of different address further supports the argument that these alerts are all generated by a single scanning application. The majority of these alerts also originated from Korean address space. It is possible that this tool is Korean in origin

Correlations

A investigation of two of the source IP addresses on www.dshield.org reveal that these addresses have been responsible for a number of attacks on port 80. Most likely using the same scanner.

IP Address: 203.233.82.121 **HostName:** 203.233.82.121 **DShield Profile:**

Country:	KR
Contact E-mail:	ip@bora.net (bounced) (bounced)
Total Records against IP:	1900
Number of targets:	581
Date Range:	2002-03-25 to 2002-03-25

Ports Attacked (up to 10):

Port	*Attacks*
80	68

Whois:

```
 IP Address : 203.233.82.0 -203.233.82.255 Connect ISP Name : BORANET
Connect Date : 20010222 Registration Date : 20010223 Network Name :
HANHWA22885D [ Organization Information ] Orgnization ID : ORG109531
Name : Hanhwa State : SEOUL Address : Hanhwa bldge 1 jangkyo  -dong Zip
Code : 100 -797 [ Admin Contact Information] Name : Chunho O Org Name
: Hanhwa State : SEOUL Address : Hanhwa bldge 1  jangkyo -dong jung -gu
Zip Code : 100 -797 Phone : +82 -2-729-4704 Fax : E -Mail :
b0022885@users.bora.net [ Technical Contact Information ] Name :
Chunho O Org Name : Hanhwa State : SEOUL Address : Hanhwa bldge 1
jangkyo -dong jung -gu Zip Code : 100 -797 Phone : +82-2-729-4704 Fax :
E-Mail : b0022885@users.bora.net
```

Evidence of active targeting

This attack has specifically targeted web servers using Internet Information Server. As the system is also scanning other hosts the attack is not specifically targeting this host. The fact that the attacker also targeted a web server with no DNS entry, and no links from external sources indicates that some form of prior reconnaissance must have taken place.

Severity

Criticality – The target system in this case is a web server hosting online courses, including student assignments, tests and results. criticality = 5

Lethality – . This attack seeks to exploit a number of vulnerabilities which would give the attacker complete control of the target system. Lethality = 5

System Countermeasures – The system was patched against all of these vulnerabilities, however given the history of this particular product there is little guarantee new vulnerabilities will not occur. The attack targets the web service, which is the primary function of the target system. This service cannot be stopped. System Countermeasures = 4

Network Countermeasures – In order to fulfil it's function the target system must be accessible to the internet on port 80. It is also not practicable to install a application firewall capable of filtering these attacks. All other ports on this system are blocked to the internet. In the event the system was compromised the attacker could only access the system through TCP port 80. Network Countermeasure s = 3

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

(5 + 5) – (4 + 3) = 3

Defensive recommendations

Maintain security updates to the system. Ensure that a virus scanner capable of detecting trojans is insta lled. Block access to all UDP and TCP ports other than 80 from outside the university network.

Multiple choice test question

When using a web browser to view pages on the web. A new port is opened on the client

a) Every time a new site is accessed

b) Everytime a new page is requested

c) after a timeout period

d) When a new browser window is opened

Correct answer **b**

Trace 4 – Packet from the broadcast address

Source of Trace

This trace ifs from a Linux system in my office connected to the staff network, via a switch. I occasionally run tcpdump on this system to see what I can find.

Detect was generated by:

tcpdump -s 512 -w 07032002.cap

Probability the source address was spoofed

The source IP address of this packet is 255.255.255.255, the global broadcast. No I P stack should be able to generate this address. The addresses is not spoofed, but the packet is crafted.

Description of attack

The only possible purpose of this attack could be to launch a Denial of Service attack. The theory would be that a invalid TCP packet from a broadcast address will elicit a response back to the broadcast address. This would then be received by everyone else who would then respond. At that point we have internet meltdown.

10:10:46.009535 255.255.255.255.31337 > MY.NET.ONE.203.51 5: R 0:3(3) ack 0 win 0

```
0x0000   4500 002b 0000 0000 0606 b9f8 ffff ffff   E..+..........
0x0010   cb0a 2fcb 7a69 0203 0000 0000 0000 0000   ../.zi........
0x0020   5014 0000 6620 0000 636b 6f00 0000        P...f...cko...
[
```

Attack mechanism

The attack is a sing le crafted packet with a source address of 255.255.255.255, the global broadcast address. This packet also has a few other unusual features.

The source port is 31337, although the person who crafted this packet is anything but eleet.

The destination port is 515, this is a print service port. The destination of this packet is a standard desktop system, running windows 98. There are no services running on port 515.

This packet also has the reset flag set, this will not elicit a response from the target, it should ignore the packet. If the attacker wanted a response he would need to use a syn. It is doubtful even then if a host would respond to 255.255.255.255

Finally the acknowledgment and window are both 0.

This packet is so far out of any standard tha t it will never be responded to.

Correlations

This traffic was mentioned on the security focus ARIS mailing list.
http://lists.jammed.com/incidents/2001/07/0025.html

Apparently packets si milar to these with the SYN flag set are also common.

Evidence of active targeting

This is a crafted packet directed towards a specific IP address. However the target does not provide any services, it is simply a desktop system. The motivation behind this attack is probably a student learning network programming, or a badly coded 'hack' tool.

Severity

Criticality – . If a target responded to this packet, and routers accepted mass broadcasts this could create a denial of service situation. As routers are unlikely to accept a destination of 255.255.255.255 and the IP stack of the target system should not respond to a tcp reset this will not occur. criticality = 1

Lethality – . Again if this attack did work and a mass broadcast could occur a broadcast storm situation would result. Lethality = 5

System Countermeasures – The IP stack of any operating system, or network device should ignore this packet System Countermeasures = 5

Network Countermeasures – Border routers will not accept traffic with a destination address of 255.255.255.255 Network Countermeasures = 5

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

(1 + 5) – (5 + 5) = -4

Defensive recommendations

No action is needed to defend against this a ttack.

Multiple choice test question

Which of the following IP addresses could be a broadcast address if CIDR is used?

a) 192.168.0.255
b) 203.54.32.239
c) 199.90.32.127
d) all of the above

Correct answer d

a is a broadcast if the mask is /24

b is a broadcast if the mask is /28

c is a broadcast if the mask is /25

Trace 5 - Traffic to the loopback address 127.0.0.1

Source of Trace

This trace was detected by a Snort system on the student network. This Snort system is located inside the NATed network.

Detect was gener ated by:

Snort

Probability the source address was spoofed

The source address may have been spoofed, however upon investigation it was found that the address had not been spoofed.

Description of attack

This attack involved multiple attempts by the attacker to connect to the destination IP 127.0.0.1. This address is defined as the loopback address and no packets addressed to or from it should be found on the network.

```
[**] [1:528:2] BAD TRAFFIC loopback traffic [**]
[Classification: Potentially Bad Traffic] [ Priority: 2]
02/15-10:08:49.884990 192.168.0.200:32889 -> 127.0.0.1:515
TCP TTL:64 TOS:0x0 ID:18958 IpLen:20 DgmLen:60 DF
******S* Seq: 0x43D8FCFF Ack: 0x0 Win: 0x16D0 TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 636657 0 NOP WS: 0

[**] [1:528:2 ] BAD TRAFFIC loopback traffic [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
02/15-10:08:52.884988 192.168.0.200:32889 -> 127.0.0.1:515
TCP TTL:64 TOS:0x0 ID:18959 IpLen:20 DgmLen:60 DF
******S* Seq: 0x43D8FCFF Ack: 0x0 Win: 0x16D0 TcpLen : 40
TCP Options (5) => MSS: 1460 SackOK TS: 636957 0 NOP WS: 0

[**] [1:528:2] BAD TRAFFIC loopback traffic [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
02/15-10:08:58.885583 192.168.0.200:32889 -> 127.0.0.1:515
TCP TTL:64 TOS:0x0 ID:1896 0 IpLen:20 DgmLen:60 DF
******S* Seq: 0x43D8FCFF Ack: 0x0 Win: 0x16D0 TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 637557 0 NOP WS: 0

[**] [1:528:2] BAD TRAFFIC loopback traffic [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
02/16-10:05:30.050498 192.168.0.200:33047 -> 127.0.0.1:515
TCP TTL:64 TOS:0x0 ID:20202 IpLen:20 DgmLen:60 DF
******S* Seq: 0x75466292 Ack: 0x0 Win: 0x16D0 TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 666744 0 NOP WS: 0

[**] [1:528:2] BAD TRAFFIC loopb ack traffic [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
02/16-10:05:33.044542 192.168.0.200:33047 -> 127.0.0.1:515
TCP TTL:64 TOS:0x0 ID:20203 IpLen:20 DgmLen:60 DF
******S* Seq: 0x75466292 Ack: 0x0 Win: 0x16D0 TcpLen: 40
TCP Options (5 ) => MSS: 1460 SackOK TS: 667044 0 NOP WS: 0

[**] [1:528:2] BAD TRAFFIC loopback traffic [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
02/16-10:05:39.045133 192.168.0.200:33047 -> 127.0.0.1:515
```

TCP TTL:64 TOS:0x0 ID:20204 IpLen:20 DgmLen:6 0 DF
******S* Seq: 0x75466292 Ack: 0x0 Win: 0x16D0 TcpLen: 40
TCP Options (5) => MSS: 1460 SackOK TS: 667644 0 NOP WS: 0

Attack mechanism

The attacker attempted to initiate a TCP connection to the host 127.0.0.1from port 33047 to port 515. On this occa sion six attempts were seen over the period of one and a half hours.

As this source address was valid for the network a quick check of the source was made. It was found to be a debian Linux system. The loopback device on this system had been disabled. B y examining the bash history it was found that the command **ifconfig lo down** had been made. (These systems are used for teaching so the students do have root access, the lab is NATed and heavily firewalled against traffic leaving it). The system used magi cfilter to enable printing to a network printer that did not support postscript printing. Once the loopback device was disabled the system started placing these packets on the network.

Correlations

Correlations were found on the firewall logs.

Evidence of active targeting

There is no evidence of active targeting, this was a case of misconfiguration.

Severity

Criticality – . This attack was only affecting student computer lab systems. The packet could not leave the network and any properly configured ro uter should ignore it
criticality = 1

Lethality – . It is unlikely that this packet would have any impact on any system. Possibly a badly implemented IP stack could accept this packet. However there is no record of this occurring. Lethality = 1

System Countermeasures – IP stacks by default will not respond to this request System Countermeasures = 5

Network Countermeasures – Routers and firewalls should be default drop these packets Network Countermeasures = 5

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

(1 + 1) – (5 + 5) = -8

Defensive recommendations

No action needed.

Multiple choice test question

How many loopback address are there?

a) 1

b) 255

c) 16581375

d) 256

correct answer c, a whole class A address space is used for the loopback address

Assignment 3 – Analyse This

Intrusion Analysis for GIAC University

Executive Summary

This analysis is based upon 7 days of log files generated by a Snort Network Intrusion Detection System (NIDS). This report provides a broad overview of security issues facing the university based solely on the information provided by this single system. It is likely that some of the alerts detected by the system are legitimate traffic, and some malicious activity has been missed by the NIDS. This report provides a starting point for further improvement of the security of the University network. It is likely that a number of systems within the University have been compromised from computer systems outside the university network. Further investigation is necessary.

The log files used in this analysis were

Alert	*Scan*	*OOS*
alert.020125	scans.020125	oos_Jan.25.2002
alert.020126	scans.020126	oos_Jan.26.2002
alert.020127	scans.020127	oos_Jan.27.2002
alert.020128	scans.020128	oos_Jan.28.2002
alert.020129	scans.020129	oos_Jan.29.2002
alert.020130	scans.020130	oos_Jan.30.2002
alert.020131	scans.020131	oos_Jan.31.2002

Overview

Three types of log file have been generated by the Snort NIDS used for this analysis. They are alerts, scans and out of scope packets. A alert is logged when traffic matching a signature pattern is detected by the NIDS. As there is a wide range of traffic traversing the network it is possible that legitimate traffic will match a signature. This will generate a alert referred to as a false positive. This is a packet that correctly matches a signature, but is not actually malicious traffic.

Over the seven day period 339,318 [1] alerts were raised. Some of these alerts were recorded many times, such as the "connect to 515 from inside" which was recorded 110,066 times. A total of 91 different types of non scan related alerts were recorded.

A scan is a information gathering exercise. In most cases the aim of a scan is to collect information to identify security weaknesses within the network. A scan may also seek to identify operating systems, services (such as web or mail servers) or network infrastructure such as proxy servers and routers. A scan in itself is not a attack on the network, it is a information gathering process. However in the majority of cases the motivation behind running a network scan is to gather information to be used in an attack. It should be noted that a scan my also be used by a network

1 Due to the high amount of scanning entries recorded in the alert log all scanning activity will be considered separately from the rest of the alerts. The figure here refers to the total number of non scan related alerts.

administrator in order to monitor the status o f the network.

A total of 112,958 individual scans were detected over the seven day period. Some of these scans were of a small number of hosts, 90,993 of the detected scans were of 10 or less hosts. However a few scans of large numbers of hosts were al so made. The largest was a scan of 1384 hosts, completed in 93 seconds by MY.NET.153.185, using 2049 UDP packets.

Comments on data collection devices

The log files provided appear to have been generated by a number of different Snort sensors, logging to a central database. This perception is supported by time inconsistencies within the log files (ie: alert times are entered out of sequence). It is likely that at least three, if not four Snort systems were used to generate these log files.

One system is using Snort the scan preprocessor to detect portscaning. When this system detects a scan occurring (when more than a defined number of ports are accesses in a defined number of seconds or a packet containing non standard flag combinations (SYN & FIN in a TCP packet for example)) it makes a entry in the alert log file. It also records the information about each packet received during the scan to a separate scan log file. During the scan it will also generate a status entry in the alert log, and once the scan has finished a summary of the scan will be recorded in the alert file.

Another system is using the http_decode preprocessor to analyse requests to web servers. This preprocessor enables a counter measure to IDS evasion techniques such as those used by whisker (RFP 2001).

It also appears that at least one sensor is located outside the border router / firewall. This is most likely the system using the port scan detection preprocessor. This system is detecting activity on ports that would normally be blocked from entering the local network. (see table 1 *Summary of scans by port)*

There is at least one system located inside the local network. This is supported by the number of detects made of traffic with a internal source and destination. (see tabl e 1) These packets should not be seen if the Snort system was located outside a firewall or border router.

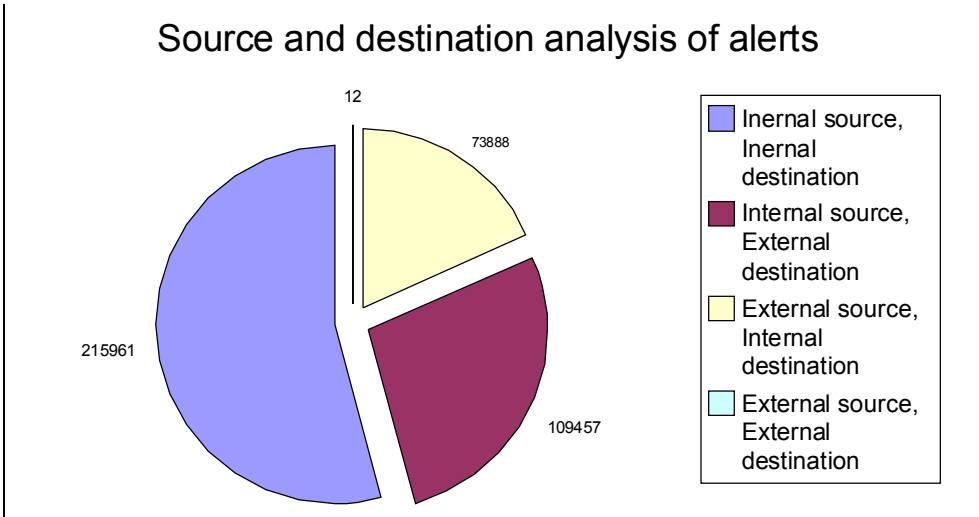


Table 1 - source and destination analysis of alerts

The most disturbing feature of this network is a low level of screening by a border router. Traffic has been detected coming into the internal network from a range unusual assigned ports and ports that would normally be blocked. Examples of this include TCP packets with source and destination ports of 137 betw een internal and external systems. Port 137 is used by the SMB protocol and provides services such as microsoft file sharing. It would be a normal expectation that this sort of traffic would be blocked by a border router or firewall. A total of 336 aler ts of this type of alert is raised with 90 different external hosts, from over 50 different networks. It is highly unlikely that these are all from legitimate sources.

Analysis of alerts

Top Talkers

Source	Total Alerts
MY.NET.70.177	17360
MY.NET.153.119	17073
63.250.209.34	10740
63.250.211.165	9859
MY.NET.153.118	9810
MY.NET.153.111	9762
63.250.209.88	9497
MY.NET.153.122	9316
MY.NET.153.114	9122
63.250.208.34	9118

Table 1Top ten sources of alerts

A brief analysis of the most common sources of alerts reveals that most of the alerts appear to be caused by hosts within the university network. Of the ten sources to generate the most alerts six of the hosts are located on the university network. T he high level of alerts from internal hosts may be caused by the location of the NIDS sensors. However this volume of alerts is a point of concern. Either the Snort systems are not configured properly and are thus generating excessive levels of false

positives, or there are a number of malicious users on the local network, or a number of local systems have been compromised. The most likely option is a combination of all three.

Destination	Total Alerts
MY.NET.150.198	110079
MY.NET.151.63	46543
MY.NET.11.6	19207
MY.NET.11.7	17156
211.115.213.202	16606
209.10.239.135	10658
MY.NET.152.109	8845
MY.NET.5.96	5378
MY.NET.11.5	4302
64.12.184.141	4031

Table 2Top ten destinations of alerts

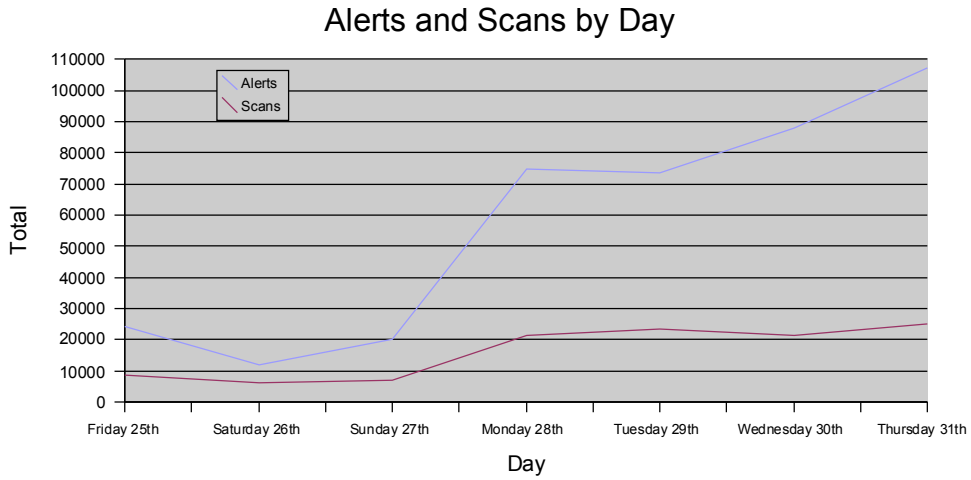
If we examine the top ten targets of attacks a similar pattern emerges. Of the top ten targets of alerts seven are local hosts. This indicates that most of the alerts are in fact targeting the local network. This traffic flow between internal and external hosts will be discussed in more detail in the next section.

A analysis of scanning activity also indicates a high level of activity from inside the network. All of the top ten scanners are from the local network. Again this may indicate malicious activity on the local network, but it is likely that these figures are corrupted by more sensors on the local network, and attempted scans being blocked by border routers.

Source	Total Scans
MY.NET.6.45	2314
MY.NET.60.43	1874
MY.NET.6.49	1690
MY.NET.6.50	1609
MY.NET.6.52	1556
MY.NET.6.60	1501
MY.NET.6.53	1441
MY.NET.6.48	1400
MY.NET.153.146	1387
MY.NET.11.6	1113

Table 3Top ten scanners

Activity over time



As can be seen from the graph the level of activity, both alerts and scans alters significantly from day to day. The low level of activity over Saturday and Sunday most likely indicates that most of the alerts are generated by local traffic. On the weekend when staff would not be using the network the level of activity drops. There is no reason why external attackers would reduce their activity over the weekend.

Analysis of Alert types

91 different types of event have been detect by the Snort system operating on the University network. Some these alert types have been recorded many times, while others have only been recorded only a few times. This section will analyse each type of detect, the number of times the detect occurred, and the implications of this detection. A brief summary of the source and destination networks of the packets triggering the detect will be made at the beginning of each analysis. This will be in the form of the number of detected packets travelling from the internal to the external network (int->ext), between hosts within the internal network (int ->int) and from external hosts to the internal network (ext ->int). The detects are ordered by the level of occurrence. In some cases, such as ICMP detects a number of detects are grouped under a common heading.

Connect to 515 from inside

Total	*int -> ext*	*int -> int*	*ext -> int*
110066	0	110066	0

As can be seen from the table above all of these detects were from internal hosts to internal hosts. In fact all but one of the *connect to 515 from inside* alerts were destined for the host MY.NET.150.198. The one other detect was destined for MY.NET.5.238. All packets originated from the MY.NET.153.0, MY.NET.152.0, MY.NET.70.O and MY.NET.88.0 subnets. This detect is designed to trigger on TCP packets destined to port 515. This port is a assigned port to network printer services. It would appear from the nature of this traffic that these detects were legitimate printer requests.

This is a unusual rule, in that it appears to be concerned with traffic from a int ernal source to a internal source, using a common service. The fact that no external hosts were detected implies that as the description of this alert indicates it does not look for external hosts connecting to TCP port 515. By it's very nature this rule is likely to raise a large number of false positives in any network using network printers. This rule could not be found in the current, or older snort rule sets available to the analyst.

There is a know exploit for the lprng print service. If MY.NET.15 0.198 is using lprng the appropriate patch should be applied. In the current ruleset there are two rules to detect attempted exploits of this vulnerability. It is suggested that removing the existing rule and replacing it with the newer rules would reduc e the number of false positives detected by the system.

New rules: (source current snort ruleset www.snort.org/download/snortrules.tar.gz)

alert TCP $EXTERNAL_NET any -> $HOME_NET 515 (msg:"EXP LOIT LPRng overflow"; flags: A+; content: "|43 07 89 5B 08 8D 4B 08 89 43 0C B0 0B CD 80 31 C0 FE C0 CD 80 E8 94 FF FF FF 2F 62 69 6E 2F 73 68 0A|";

reference:bugtraq,1712; classtype:attempted -admin; sid:301; rev:1;)

alert TCP $EXTERNAL_NET any -> $HOME_NE T 515 (msg:"EXPLOIT redhat 7.0 lprd overflow"; flags: A+; content:"|58 58 58 58 25 2E 31 37 32 75 25 33 30 30 24 6E|"; classtype:attempted -admin; sid:302; rev:1;)

spp_http_decode: IIS unicode attack detected

Total	*int -> ext*	*int -> int*	*ext -> int*
89076	88866	66	144

This alert detects when http requests contain unicode, rather than plain ASCII as a component of the requested URL. This is commonly used to defeat both IDSes and protection on the web server. In some cases susceptible web servers will allow clients to access files outside the web directories.

As can be see from the summaries most of these detects are from internal hosts accessing external hosts. This may be a indication of malicious activity by internal users. It is also likely that a po rtion of these alerts are false positives as some web clients translate some characters, especially spaces into unicode in order to access directory and file names containing spaces, or other characters not usually see by a particular system. Netscape in particular will do this.

15 local hosts have been targeted by requests containing unicode. This requests can only by effective if the hosts are running a web server and are susceptible to the particular attack. Each of the internal systems should be ch ecked, and if they are running web servers the appropriate patches should be applied. If they are not running web servers the border router should be configured to block access from external hosts to these addresses. The internal hosts are:

MY.NET.11.4
MY.NET.150.107
MY.NET.150.16
MY.NET.150.198
MY.NET.150.6
MY.NET.150.83
MY.NET.151.114
MY.NET.153.219
MY.NET.5.141
MY.NET.5.249
MY.NET.5.92
MY.NET.5.95
MY.NET.5.96
MY.NET.5.97
MY.NET.88.190

A number of attempts were made by 211.90.176.59 to use various web server exploits against three of these web servers (150.107, 150.16 and 5.249) on the 30th and 31st of January. There is no indication that any of these attempts were successful.

MY.NET.150.198 is the print server identified in the previous detect. It was also the target of various web server exploit attempts, all from 61.75.72.2

MY.NET.150.6 was targeted by 200.64.239.148 with what appears to be the same selection of exploits as used in the previous two attempts.

MY.NET.150.83 was the target of a range of attempted exploits and is possibly compromised. Although it is more likely the detected trojan activity is a false positive. For more detail see *Possible trojan server activity* below.

MY.NET.151.114 and MY.NET.153.219 were the target of a range of web server exploits

MY.NET.5.249 and MY.NET.5.141, MY.NET.5.95, MY.NET.5.96, MY.NET.5.97, MY.NET.88.190 - attempts by multiple hosts to exploit web server weakness. All of these systems are also app ear to be using SNMP and SMB. See comments in recommendations at the end of this document.

misc large UDP packet

Total	*int -> ext*	*int -> int*	*ext -> int*
53791	0	0	53791

Large UDP packets are not commonly seen on TCP/IP networks. Possible uses of large UDP packets are denial of service attacks, near real time applications, networked games, peer to peer file sharing applications such as GNUnet or streaming video or audio.

Snort Rule: alert UDP $EXTERNAL_NET any -> $HOME_NET any (msg:"MISC Large UDP Pack et"; dsize: >4000; reference:arachnids,247; classtype:bad-unknown; sid:521; rev:1;)

The rule generating this alert is only set to log packets from external sources to internal sources. No information is collected on possible responses to these large UDP packets. The minimum size of a UDP packet is 4000bytes.

If this traffic was due to network games (quite possible in a university network) it would be expected that large packets would also be transmitted from the local machines. This could be tested by altering the rule to also log traffic from the internal network to the external network.

The wide variance in the ports used in all these cases implies that the applications generating this traffic are peer to peer based rather than client server. In a client server relationship a fixed port would usually be used by the server.

Denial of service attacks are a possibility, however the duration of this traffic is quite long. The first of these packets between 63.250.209.34 and MY.NET.151.63 is logged at 25/018:56:56. This section of alerts then stops at 11:54:54 on the same day. The alerts begin again at 08:51:24 on the 28th and continues for more than 24hrs until later at 10:45:47 on the 29th. It would be hoped that if this was a successful denial of service attack it would have been noticed and filtering would have been put in place within 24 hours at the most. However it is possible that these alerts relate to a attempted distributed denial of service attack.

Another peculiarity is the connection between the host network of the source addresses. The packets directed towards MY.NET.151.63 all originate from sources in the 63.250 address space. The packets directed towards MY.NET.153.185 all originate from the 211.233.70 address space. In all cases multiple alerts are raised between different source networks and individual local hosts.

In order to gain further

Total packets	Source	sport	Destination	dport
4	68.55.200.56	7000	MY.NET.150.143	7001
1	207.25.79.241	2969	MY.NET.150.79	2730
9118	63.250.208.34	2383	MY.NET.151.63	3276
1	63.250.209.162	3908	MY.NET.151.63	2036
10740	63.250.209.34	3694	MY.NET.151.63	3236
302	63.250.209.74	3103	MY.NET.151.63	1462
9495	63.250.209.88	1672	MY.NET.151.63	4314
7246	63.250.210.50	1396	MY.NET.151.63	1518
9859	63.250.211.165	4783	MY.NET.151.63	4330
142	210.218.249.23	3404	MY.NET.153.113	2455
199	211.233.27.142	38568	MY.NET.153.160	16896
185	211.233.70.161	1730	MY.NET.153.185	2105
1028	211.233.70.162	2568	MY.NET.153.185	2072
354	211.233.70.163	1060	MY.NET.153.185	2331
193	211.233.70.165	3338	MY.NET.153.185	2426
223	211.233.70.172	2966	MY.NET.153.185	2123
840	202.58.33.70	3058	MY.NET.153.191	1388
27	63.250.208.38	0	MY.NET.153.193	0
677	64.152.216.77	54540	MY.NET.153.194	2170
3109	203.231.232.15	3450	MY.NET.153.195	2327
47	63.250.211.197	0	MY.NET.153.210	0
1	207.25.79.240	7001	MY.NET.88.181	7000

Illustration 2Large UDP traffic by source and destination

information about this traffic a search of alerts with a source of MY.NET.151.63 was made. There were 1565 "ICMP Fragment Reassembly Time exceeded" messages sent by MY.NET.151.63 to 63.250.208.34. This would possibly indicate a denial of service attack. The attacker could have sent UDP headers, indicating a large fragmented packet. The common maximum size of packets is 1500bytes if the hosts are on a ethernet network, given the dominance of ethernet as a LAN network protocol this is a reasonable assumption. The destination would then store the large packets in a buffer waiting for the rest of the fragments, which would never be sent. However these ICMP messages were only by MY.NET.151.63 to 63.250.208.34. No other ICMP Fragment Reassembly Time Exceeded messages have been detected, despite the large number of other large UDP packets detected.

Further examination of the flow of these alerts show that the packets arrived at a rate of up to 3 packets per second. The source and destination port would also change with no change in the flow rate of packets. This is extremely unusual behaviour and as it occurs on a number of occasions introduces the possibility that the source address is spoofed.

2002-01-30 08:51:24	MISC Large UDP Packet	63.250.211.165	1809 MY.NET.151.63	4398
2002-01-30 08:51:24	MISC Large UDP Packet	63.250.211.165	1809 MY.NET.151.63	4398
2002-01-30 08:51:24	MISC Large UDP Packet	63.250.211.165	1809 MY.NET.151.63	4398
2002-01-30 08:51:25	MISC Large UDP Packet	63.250.211.165	1809 MY.NET.151.63	4398
2002-01-30 08:51:25	MISC Large UDP Packet	63.250.211.165	1809 MY.NET.151.63	4398
2002-01-30 08:51:25	MISC Large UDP Packet	63.250.211.165	1809 MY.NET.151.63	4398
2002-01-30 08:51:26	MISC Large UDP Packet	63.250.211.165	1926 MY.NET.151.63	4403
2002-01-30 08:51:26	MISC Large UDP Packet	63.250.211.165	1926 MY.NET.151.63	4403
2002-01-30 08:51:26	MISC Large UDP Packet	63.250.211.165	1926 MY.NET.151.63	4403
2002-01-30 08:51:27	MISC Large UDP Packet	63.250.211.165	1926 MY.NET.151.63	4403
2002-01-30 08:51:27	MISC Large UDP Packet	63.250.211.165	1926 MY.NET.151.63	4403

Illustration 3Note change in sport and dport, but no change in flow rate

However a check of the registered owner of the 63.250.209.74 address indicates that it is registered to yahoo broadcast services

> Yahoo! Broadcast Services, Inc.
>
> (NETBLK-NETBLK2-YAHOOBS) 2914 Taylor st Dallas, TX 75226
> US Netname: NETBLK2-YAHOOBS Netblock: 63.250.192.0 - 63.250.223.255
> Maintainer: YAHO Coordinator: Bonin, Troy (TB501 -ARIN)
> netops@broadcast.com 214.782.4278 ext. 2278
>
> Domain System inverse mapping provided by:
> NS.BROADCAST.COM 206.190.32.2
> NS2.BROADCAST.COM 206.190.32.3
>
> ADDRESSES WITHIN THIS BLOCK ARE NON-PORTABLE
> Record last updated on 29-Jun-2001. Database last updated on 11-Mar-2002 19:58:33 EDT. The ARIN Registration Services Host contains ONLY Internet Network Information: Networks, ASN's, and related POC's. Please use the whois server at rs.internic.net for DOMAIN related Information and whois.nic.mil for NIPRNET Information.

This would make it highly probable that the majority of this traffic has been generated by streaming media.

SMB Name wildcard

Total	*int -> ext*	*int -> int*	*ext -> int*
40983	0	40630	353

This alert identifies attempts by a hosts to identify shared direc tories on a windows system. This type of traffic is extremely common in networks using microsoft windows operating systems. Traffic between internal hosts originating from port 137 and destined to port 137 is most likely to be false positives. Traffic f rom external sources to local hosts may be attempts to connect to unsecured shared directories. It is good policy to ensure that these ports are blocked by a firewall. For further information see http://www.sans.org/y2k/050300.htm. Of greater concern are the alerts originating from ports other than 137. There are 18 of these alerts. These are listed below.

cap_date	descrip	src	sport	dst	dport
2002-01-29 06:41:01	SMB Name Wildcard	MY.NET.144.14	32791	MY.NET.5.74	137
2002-01-31 13:17:51	SMB Name Wildcard	24.232.104.105	1024	MY.NET.150.240	137
2002-01-31 15:10:01	SMB Name Wildcard	62.110.204.187	3394	MY.NET.88.162	137
2002-01-31 15:10:05	SMB Name Wildcard	62.110.204.187	3394	MY.NET.88.162	137
2002-01-31 15:10:18	SMB Name Wildcard	62.110.204.187	3394	MY.NET.88.162	137
2002-01-31 15:22:20	SMB Name Wildcard	62.110.204.187	3394	MY.NET.88.162	137
2002-01-31 15:22:54	SMB Name Wildcard	62.110.204.187	3394	MY.NET.88.162	137
2002-01-31 16:31:14	SMB Name Wildcard	200.38.214.180	322	MY.NET.150.133	137
2002-01-31 18:44:12	SMB Name Wildcard	63.203.5.6	24	MY.NET.150.133	137
2002-01-31 18:44:13	SMB Name Wildcard	63.203.5.6	24	MY.NET.150.133	137
2002-01-31 18:44:15	SMB Name Wildcard	63.203.5.6	24	MY.NET.150.133	137
2002-01-31 20:28:06	SMB Name Wildcard	216.248.137.107	1025	MY.NET.88.162	137
2002-01-31 20:28:06	SMB Name Wildcard	216.248.137.107	1026	MY.NET.88.162	137
2002-01-31 20:28:07	SMB Name Wildcard	216.248.137.107	1026	MY.NET.88.162	137
2002-01-31 20:42:00	SMB Name Wildcard	65.94.48.68	1231	MY.NET.88.162	137
2002-01-31 20:48:24	SMB Name Wildcard	65.30.100.27	20705	MY.NET.88.162	137
2002-01-31 21:08:17	SMB Name Wildcard	195.24.22.129	819	MY.NET.88.162	137
2002-01-31 21:08:19	SMB Name Wildcard	195.24.22.129	819	MY.NET.88.162	137

This list shows a number of attempts by mostly external hosts to access information about SMB shares on local hosts. This activity is most likely malicious and deserves further attention.

This type of access should be blocked by a firewall, or at least a border router.

SNMP Public Access

Total	*int -> ext*	*int -> int*	*ext -> int*
33800	0	33796	4

This alert is of extreme concern. As the name (Simple Network Management Protocol) implies SNMP is used for management and configuration of network devices. This protocol is used by a wide range of manufactures and products. In order to provide a level of security version 1 of the protocol implemented a simple string authentication mechanism. If you know the right string you can access the device. By default this string is set to public.

There are also a large number of vulnerabilities. that have rec ently been discovered with this protocol see http://www.kb.cert.org/vuls/id/107186 for more details.

ICMP Echo Request L3retriever Ping

Total	*int -> ext*	*int -> int*	*ext -> int*
20236	0	20217	19

Total	*int -> ext*	*int -> int*			*ext -> int*		
description	count(src)	source	:	sport	destination	:	dport
connect to 515 from inside	1662	MY.NET.152.166	:	3834	MY.NET.150.198	:	515
connect to 515 from inside	1526	MY.NET.152.158	:	1854	MY.NET.150.198	:	515
connect to 515 from inside	1351	MY.NET.152.161	:	1259	MY.NET.150.198	:	515
connect to 515 from inside	1151	MY.NET.152.22	:	1374	MY.NET.150.198	:	515
connect to 515 from inside	1032	MY.NET.152.167	:	2203	MY.NET.150.198	:	515
connect to 515 from inside	989	MY.NET.152.169	:	1311	MY.NET.150.198	:	515
connect to 515 from inside	974	MY.NET.152.180	:	1294	MY.NET.150.198	:	515
connect to 515 from inside	898	MY.NET.152.162	:	1465	MY.NET.150.198	:	515
connect to 515 from inside	871	MY.NET.152.182	:	1299	MY.NET.150.198	:	515
connect to 515 from inside	708	MY.NET.152.164	:	1712	MY.NET.150.198	:	515
High port 65535 udp - possible Red Worm - traffic	1380	MY.NET.152.183	:	65535	MY.NET.6.49	:	20712
High port 65535 udp - possible Red Worm - traffic	1166	MY.NET.152.170	:	255	MY.NET.6.52	:	65535
High port 65535 udp - possible Red Worm - traffic	946	MY.NET.152.186	:	65535	MY.NET.6.60	:	65535
High port 65535 udp - possible Red Worm - traffic	903	MY.NET.152.160	:	65535	MY.NET.6.52	:	65535
High port 65535 udp - possible Red Worm - traffic	634	MY.NET.152.173	:	65535	MY.NET.6.49	:	65535
High port 65535 udp - possible Red Worm - traffic	540	MY.NET.152.11	:	43263	MY.NET.6.53	:	65535
ICMP Echo Request L3retriever Ping	1984	MY.NET.152.21	:	0	MY.NET.11.6	:	0
ICMP Echo Request L3retriever Ping	1605	MY.NET.152.46	:	0	MY.NET.11.7	:	0
ICMP Echo Request L3retriever Ping	1161	MY.NET.152.159	:	0	MY.NET.11.6	:	0
ICMP Echo Request L3retriever Ping	1095	MY.NET.152.172	:	0	MY.NET.11.7	:	0
ICMP Echo Request L3retriever Ping	1040	MY.NET.152.215	:	0	MY.NET.11.6	:	0
ICMP Echo Request L3retriever Ping	1027	MY.NET.152.213	:	0	MY.NET.11.7	:	0
ICMP Echo Request L3retriever Ping	855	MY.NET.152.163	:	0	MY.NET.11.7	:	0
ICMP Echo Request L3retriever Ping	841	MY.NET.152.44	:	0	MY.NET.11.7	:	0
ICMP Echo Request L3retriever Ping	802	MY.NET.152.19	:	0	MY.NET.11.7	:	0
ICMP Echo Request L3retriever Ping	773	MY.NET.152.250	:	0	MY.NET.11.7	:	0
ICMP Echo Request L3retriever Ping	744	MY.NET.152.175	:	0	MY.NET.11.6	:	0
ICMP Echo Request L3retriever Ping	737	MY.NET.152.13	:	0	MY.NET.11.6	:	0
ICMP Echo Request L3retriever Ping	736	MY.NET.152.174	:	0	MY.NET.11.7	:	0
ICMP Echo Request L3retriever Ping	718	MY.NET.152.246	:	0	MY.NET.11.7	:	0
ICMP Echo Request L3retriever Ping	697	MY.NET.152.247	:	0	MY.NET.11.6	:	0
ICMP Echo Request L3retriever Ping	668	MY.NET.152.16	:	0	MY.NET.11.7	:	0
ICMP Echo Request L3retriever Ping	662	MY.NET.152.245	:	0	MY.NET.11.6	:	0
ICMP Echo Request L3retriever Ping	630	MY.NET.152.244	:	0	MY.NET.11.6	:	0
ICMP Echo Request L3retriever Ping	628	MY.NET.152.185	:	0	MY.NET.11.7	:	0
ICMP Echo Request L3retriever Ping	626	MY.NET.152.214	:	0	MY.NET.11.7	:	0
ICMP Echo Request L3retriever Ping	541	MY.NET.152.14	:	0	MY.NET.11.6	:	0
ICMP Echo Request L3retriever Ping	526	MY.NET.152.12	:	0	MY.NET.11.6	:	0
ICMP Echo Request L3retriever Ping	359	MY.NET.152.249	:	0	MY.NET.11.7	:	0
ICMP Echo Request L3retriever Ping	336	MY.NET.152.17	:	0	MY.NET.11.6	:	0
ICMP Echo Request L3retriever Ping	294	MY.NET.152.10	:	0	MY.NET.11.6	:	0
ICMP Echo Request L3retriever Ping	73	MY.NET.150.86	:	0	MY.NET.5.4	:	0
ICMP traceroute	1574	MY.NET.152.171	:	0	MY.NET.152.1	:	0
INFO MSN IM Chat data	4252	MY.NET.152.179	:	1544	64.4.12.174	:	1863
INFO MSN IM Chat data	1361	MY.NET.152.157	:	2322	64.4.12.173	:	1863
INFO MSN IM Chat data	1159	MY.NET.152.177	:	1226	64.4.12.182	:	1863
INFO MSN IM Chat data	1142	MY.NET.152.184	:	2588	64.4.12.195	:	1863
INFO MSN IM Chat data	1070	MY.NET.152.181	:	1653	64.4.12.154	:	1863
INFO MSN IM Chat data	541	MY.NET.152.178	:	1607	64.4.12.158	:	1863
SMB Name Wildcard	947	MY.NET.152.251	:	137	MY.NET.237.186	:	137
SMB Name Wildcard	797	MY.NET.152.216	:	137	MY.NET.224.106	:	137
SMB Name Wildcard	699	MY.NET.152.15	:	137	MY.NET.237.186	:	137
SMB Name Wildcard	675	MY.NET.152.18	:	137	MY.NET.222.66	:	137
SMB Name Wildcard	648	MY.NET.152.248	:	137	MY.NET.222.66	:	137
SMB Name Wildcard	541	MY.NET.152.168	:	137	MY.NET.224.106	:	137
SMB Name Wildcard	246	MY.NET.152.252	:	137	MY.NET.210.78	:	137
SNMP public access	1125	MY.NET.150.179	:	1125	MY.NET.150.198	:	161
spp_http_decode: IIS Unicode attack detected	1778	MY.NET.152.165	:	1133	MY.NET.11.4	:	80
spp_http_decode: IIS Unicode attack detected	1347	MY.NET.152.176	:	2020	205.188.180.25	:	80

alert icmp any any -> $HOME_NET any (msg:"ICMP L3retriever Ping"; content: "ABCDEFGHIJKLMNOPQRSTUVWABCDEFGHI"; itype: 8; icode: 0; depth: 32; reference:arachnids,311; classtype:attempted -recon; sid:466; rev:1;)

This alert set to match any ICMP echo request packet containing the string "ABCDEFGHIJKLMNOPQRSTUVWABCDEFGHI". This is characteristic of a ping packet used by the L3 Retriever security tool. The m ajority of detects show communication between local hosts, this conforms to the manner in which it is expected the tool would be used.

Only four external hosts have used this scan, all of them targeting the web server MY.NET5.96. 15 of these packets origi nating from the single source 68.55.192.95.

The level of usage from a local addresses is much higher. The majority of this type of traffic was travelling from MY.NET.150.0/24 to MY.NET.11.7. This would indicate that a number of copies of this tool are i nstalled on different hosts within MY.NET.150.0/24. If the security officer for this network has not installed this tool further investigation should be carried out.

Further reference: http://enterprisesecurity.symantc.com

spp_http_decode: CGI Null Byte a ttack detected

Total	*int -> ext*	*int -> int*	*ext -> int*
11338	11152	174	12

This alert indicates attempted attacks against a web server. In this case the attacker has attempted to exploit a cgi script by passing a request for a file with a null byte on the end of the request. This alert is designed to detect the NULL byte. This exploit may allow attackers to access (read) files on the web server.

http://www.cve.mitre.org/cgi -bin/cvename.cgi?name=CVE -2000-0332

In order to determine if this attack was successful a search was made to find if any target hosts of this attack have been the source of any further alerts. Only two local hosts were targeted by this attack, MY.NET.5.95 a nd MY.NET.5.96. There are no alerts with a source of MY.NET.5.95. There are 316 alerts with a source of MY.NET.5.96. However these are all either common false positives, such as the SMB Name wildcard, from port 137 to port 137 within MY.NET, or WEB MISC 304 Forbidden. These are common false positives when dealing with a Microsoft IIS web server.

There are however two alerts of concern, *possible myserver activity* and *Backdoor Netmetro file list*. Both of these alerts relate to trojan type activity. Ho wever these alerts are based upon identifying the TCP ports used by these tools. This is a alert technique prone to false positives. In every case the alerts relating to possible myserver activity occur over a short period of time with a single IP addres s and the same IP does not occur again in the logs. This activity is consistent with a standard websession where the client happens to be using the ports also used by a trojan. If the ports were used again it might be more of a case for alarm. The other point is that the ports identified are normally used on the server side of the trojans. If this system was compromised we would see activity with it as the destination rather than the source.

A analysis of all alerts destined for MY.NET.5.95 and MY.NET .5.96 does not reveal any further concerning information. A number of attempts have been made to

compromise the system, however there is no evidence of a successful compromise.

High port 65535 UDP - possible Red Worm - traffic & High port 65535 TCP - possible Red Worm - traffic

	Total	*int -> ext*	*int -> int*	*ext -> int*
UDP	7658	6	5821	1831
TCP				

The Red worm, also known as Adore is a Linux based worm first identified in march 2001. The worm exploits four known weaknesses in common Linux services. For more details see http://www.sans.org/y2k/adore.htm. At the time of this alert the worm was over 8 months old and all Linux versions have patches available to fix the security loop holes the worm uses. So the chances of a system being infected is relatively low.

One of the vulnerabilities that this worm seeks to exploit is a weakness in the LPRng print server, operating on port 515, however none of these alerts have connections to or from 515.

The fact that this alert is only based upon the use of a particular port increases the likelihood of false positives.

sport	count(sport)	dport
65535	5739	65535
57599	285	65535
55551	199	65535
61695	142	65535
43263	127	65535
62975	97	65535
59647	86	65535
255	83	65535
41215	76	65535
16383	73	65535

Table 5Count of source port to destination port of UDP high port traffic

Another trojan that uses this port is the RC1 Trojan. This is a windows based trojan, written in Visual Basic. So although the message associated for this alert is focused towards Red Worm traffic, the alert will also detect RC1 traffic. This eliminates identifying the operating system on source and destination addresses in order to identify if the alert is relevant or not.

One interesting pattern in these alerts is the large amount (5739 connections) of UDP traffic from port 65535 to port 65535 (see table). This far exceeds the next level of traffic (285 connections), from port 57599 to 65535.

The fact that none of the traffic to or from port 65535 also connects to makes the probability of a machine infected with the Red worm unlikely.

There is also 13 alerts relating to traffic from UDP port 0 to port 65535. This is highly unusual traffic and should be investigated further. The list of systems sending this data is listed below.

count(src)	src	sport	dst	dport
5	MY.NET.6.48	0	MY.NET.153.189	65535
4	MY.NET.6.52	0	MY.NET.153.173	65535
2	MY.NET.6.49	0	MY.NET.153.202	65535
1	MY.NET.6.50	0	MY.NET.153.172	65535
1	202.229.62.134	0	MY.NET.88.137	65535

The RC1 trojan cannot be eliminated as a possible source of the high port connections. The unusual traffic with t he same source and destination port is also worthy of further investigation. Reports from IANA, www.snort.org and www.dshield.org do not have any indication of malicious traffic using these ports, nor any reference to applications that would commonly use these ports. The top 10 sources of this traffic are listed below.

count(src)	src	sport	dst	dport
896	MY.NET.6.49	65535	MY.NET.153.163	65535
807	MY.NET.6.48	65535	MY.NET.153.189	65535
778	MY.NET.6.52	65535	MY.NET.153.179	65535
751	MY.NET.6.50	65535	MY.NET.153.176	65535
281	64.152.108.142	65535	MY.NET.88.163	65535
172	64.152.108.141	65535	MY.NET.88.163	65535
160	63.210.134.141	65535	MY.NET.88.163	65535
110	MY.NET.6.60	65535	MY.NET.153.181	65535
90	MY.NET.6.53	65535	MY.NET.151.191	65535
74	MY.NET.6.45	65535	MY.NET.153.163	65535

INFO MSN IM Chat Data

This alert indicates users using the microsoft chat application. This traffic does not constitute a threat to network security.

Watchlist 000220 IL-ISDNNET-990517

This alert relates to a warning issued by NIPC about attacks from some middle eastern networks one of which was IL -ISDNNET. http://www.incidents.org/archives/y2k/110200.htm This warning was is sued in October 2000 and is no longer relevant. This rule could be removed from the Snort ruleset. (It has in fact been removed from the current ruleset)

ICMP echo request Nmap or HPing2

Nmap and Hping2 are both scanning tools used for identifying host o perating systems. Both these tools are used extensively on the internet. This traffic is information gathering rather than malicious. However the information gathered may be used as the basis of a attack at a later date. One useful correlation is to co mpare sources of these scans and relate them back to the number of attack attempts from the same source. A list of hosts that scanned using Nmap or Hping2 and were the source of other alerts is shown below.

This list shows a number of interesting things. Firstly the majority of scanning is from local hosts. This may indicate compromised hosts, or more likely students experimenting with various scanning tools. The fact that only three hosts are actively launching attacks leads to the conclusion that the majority of these scans to not have malicious intent. The three hosts that launch IIS unicode exploits should be investigated further. (as mentioned earlier).

ICMP:

Alert	**Total**
ICMP Destination Unreachable (Host Unreachable)	2561
ICMP Router Selection	2495
ICMP Destination Unreachable (Communication Administratively Prohibited)	2033
ICMP Fragment Reassembly Time Exceeded	1565
ICMP Echo Request Windows	821
ICMP Echo Request BSDtype	665
ICMP Source Quench	279
ICMP Destination Unreachable (Network Unreachabl e)	116
ICMP traceroute	43
ICMP Destination Unreachable (Fragmentation Needed and DF bit was set)	42
ICMP Destination Unreachable (Protocol Unreachable)	38
ICMP redirect (Host)	27
ICMP Address Mask Reply	6
ICMP Echo Request CyberKit 2.2 Windows	5
ICMP SRC and DST outside network	2
ICMP Address Mask Request	2
ICMP Echo Request Delphi -Piette Windows	2

Total	***int -> multicast***	***int -> ext***	***int -> int***	***ext -> int***	***ext -> ext***
10702	2495	1603	1732	4870	2

Internet Control Message Protocol messages are used to commun icate control information between hosts using IP. All of these messages can have a legitimate reason for being generated. However in some cases these alerts may either provide information to potential attackers, or indicate a failure by a attacker to acc ess the local network.

In some cases packets can be constructed to generate ICMP messages thus allowing a user to develop a image of the network. ICMP messages likely to be caused by probes are Destination Unreachable (fragmentation needed and DF bit w as set),

traceroute, and protocol unreachable.

The analyst should also be aware that ICMP packets are generally a response to another event. Thus the destination of a ICMP packet is in fact the potential attacker.

The summary of traffic flow above includes the field multicast. This is caused by the Router Selection requests which are multicast to the internal network. This is normal network traffic and these packets should not leave the local network unless a router is misconfigured.

If a attacker is probing a network in order to elicit ICMP responses it is likely that different ICMP responses from different hosts would be seen. The is no traffic of this type between internal and external hosts.

Web server Alerts

These alerts all relate to traffic to and from a web server. These alerts can be divided into two types, alerts generated by traffic from the server, attempts to exploit known vulnerabilities on the server.

Alerts generated from a web server:

WEB-MISC 403 Forbidden 50

This alert simply indicates that a attempt has been made by a user to access a page on the web server for which they do not have the correct permission. This can also be generated if a user enters the wrong username or password and then clicks the cancel button on the dialog box generated by their web browser. Corresponding entries should be made in the web server log files.

This alert is particularly useful in this analysis as it provides the analyst with positive identification of web servers in the local network. A summary of these alerts, listed by web server is shown below.

count(src)	source	sport	destination	dport
11	MY.NET.150.101	:80	68.55.205.25	:3859
1	MY.NET.150.198	:80	61.75.72.2	:2441
6	MY.NET.5.141	:80	203.227.74.100	:15081
8	MY.NET.5.92	:80	64.152.75.2	:47300
24	MY.NET.5.96	:80	208.242.127.35	:1411

Alerts generated to a web server:

These alerts are all triggered by specific content in a http request. As a result of this they have a low level of false positives. In the majority of cases they indicate a attempt to exploit a known weakness in a web server. Having said this the previous alerts identified 5 web servers on the internal network. All of these machines have been identified a running a Microsoft operating system and are most likely using Internet Information Server. This reduces the relevance of some of the alerts detected which are targeting Apache or Netscape web servers.

WEB-IIS view source via translate header 1244
This alert indicates that a user is attempting to view the source code of a .asp or .htr file on the web server. This files contain scripts used to dynamically serve WebPages. If a attack can obtain the source code for these scripts the code can then be examined for weaknesses, most commonly buffer overflows. For furt her reference see: http://www.securiteam.com/windowsntfocus/5LP0D2A2AW.html

This exploit was first identified in August 2000, and a patch has been available since that time. It is recommended that the server administrator confirms that all patches have been applied.

These exploits were focused on two local servers the majority of attempts (1234) were targeted at MY.NET.5.96. The remaining 10 attacks were targeting MY.NET.150.220.

WEB-IIS _vti_inf access 323
This alert indicates a test by a user to see if Microsoft Front -page extensions are enabled on a web server. To trigger this alert the user has simple tried to download a file called _vti_inf.html. If this file exists then Mirco soft Front-page extensions have been installed on the server. There are a number of vulnerabilities with these extensions. All 323 of these alerts targeted MY.NET.5.96 and were generated by 95 different external hosts. For more details see http://www.mobrien.com/windows_nt_wardoc.htm

WEB-FRONTPAGE _vti_rpc access 298
This alert also relates to the Microsoft front -page extensions. Again the only targeted local host is MY.NET.5.96. A total of 93 different sources triggered this alert, of those 93, 89 also triggered the _vti_inf alert. This introduces the possibility that a scanner is being used to locate vulnerable web servers. For more details see: Bugtraq ID 2144 http://online.securityfocus.com/bid/2144

WEB-MISC Attempt to execute cmd 283
This alert indicates that a http request was received that contained the string "cmd.exe". This is the command shell on a Windows NT/2000 server. At tempts to access cmd.exe are normally made in order to execute commands on the local system. Rather than open a remote shell as might be possible on a UNIX type system. The 14 systems listed below were the targets of this attack.

No hosts that triggered this alert triggered the front-page extensions alerts.

Destination	count(alert.dst)
MY.NET.5.96	72
MY.NET.5.141	32
MY.NET.5.249	28
MY.NET.5.95	27
MY.NET.150.83	24
MY.NET.88.190	16
MY.NET.150.16	14
MY.NET.151.114	14
MY.NET.5.92	13
MY.NET.5.97	13
MY.NET.150.107	9
MY.NET.153.219	8
MY.NET.150.198	8
MY.NET.150.6	5

WEB-CGI scriptalias access **90**

This alert indicates an attempt to exploit a vulnerability. in Apache and NCSA web severs that allows the attacker to view the source code of scripts on the web server. This is the same problem discussed earlier with the *WEB-IIS view source via translate header* alert. As can be seen from the list below these attacks are all directed towards MY.NET.5.96. This server appears to be running a Microsoft operating system and IIS, so any attacks attempting to exploit a vulnerability. in Apache or NSCA are unlikely to be successful.

count	source	destination
35	162.33.219.85	MY.NET.5.96
24	12.26.86.13	MY.NET.5.96
11	62.104.214.104	MY.NET.5.96
6	162.33.219.99	MY.NET.5.96
3	64.192.55.25	MY.NET.5.96
2	204.192.48.137	MY.NET.5.96
2	68.55.193.163	MY.NET.5.96
1	199.228.142.3	MY.NET.5.96
1	151.196.167.115	MY.NET.5.96
1	68.54.201.254	MY.NET.5.96
1	131.118.250.212	MY.NET.5.96
1	198.22.121.120	MY.NET.5.96
1	24.163.18.148	MY.NET.5.96
1	162.33.219.58	MY.NET.5.96

None of these source addresses were detected attempting the previous web server exploits.

For more detail on this vulnerability see:

http://online.securityfocus.com/bid/2300

http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0236

WEB-MISC Compaq insight dire ctory traversal **63**

Despite the name of this alert it is not specific to Compaq Insight systems. This alert indicates a user attempting to use "../" to access files outside the root of the web directory. A number of web servers have been vulnerable to th is type of attack including early versions of Microsoft IIS. However this exploit is now approaching 3 years old and it would be highly unusual for it to be effective against a current web server. It is interesting to note that none of these alerts relat e to the web servers targeted by the attacks mentioned earlier.

count (src)	source	destination
28	64.29.223.152	MY.NET.153.189
14	164.109.145.90	MY.NET.153.126
5	216.194.84.100	MY.NET.153.211
5	165.193.152.71	MY.NET.153.120
2	64.152.4.80	MY.NET.153.209
1	66.28.44.148	MY.NET.153.188
1	64.70.76.144	MY.NET.153.166
1	64.14.118.196	MY.NET.153.186
1	63.211.210.22	MY.NET.153.177
1	139.142.105.56	MY.NET.150.165
1	128.220.240.54	MY.NET.153.122
1	128.164.127.252	MY.NET.152.179
1	131.118.254.39	MY.NET.152.170
1	128.8.10.189	MY.NET.153.146

For more details on this vulnerability. see:

http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0771

WEB-MISC http directory traversal 35

This alert is identical to the previous one. It is assumed that this alert is from a different Snort system. It should be noted that the current Snort ruleset includes both of these alerts. However they both look for exactly t he same information (flags: A+; content: "../";)

As can be seen from the alert summary all of these attacks target MY.NET.5.96. This would indicate that the previous alert was generated by a different Snort system, on a different LAN.

count(alert.src)	src	dst
26	12.26.86.13	MY.NET.5.96
6	151.196.167.115	MY.NET.5.96
1	68.33.52.106	MY.NET.5.96
2	68.54.201.254	MY.NET.5.96

WEB-MISC whisker head

Whisker is a cgi scanning tool developed by Rain Forest Puppy. Whisker is designed to scan web servers looking for known vulnerabilities. Whisker also uses a number of NIDS evasion techniques such as fragmentation. For more details on whisker see http://www.wiretrip.net/rfp/pages/whitepaper/whiskerids.html . This alert is triggered when a http packet containing "HEAD/./" is found. This is an attempt to elude a IDS scanning http packets, looking for GET r equests. Rather than using GET whisker uses HEAD (this is still a valid http request). Whisker is a common tool freely available on the internet so scans be whisker cannot be considered uncommon, or surprising. In this case all 16 detects have a source address of 12.91.164.96 and destination address of MY.NET.5.96. All these alerts occur over a period of 2 minutes. This may indicate that the scanner is probably scanning a number of other servers at the same, or that other scans are also being made whic h have not been detected. The fact that whisker is a moderately difficult tool to use and is specifically designed to avoid intrusion detection systems would justify checking on other activity from 12.91.164.96.

WEB-CGI formmail access 8

Formmail is a cgi script written in Perl for enabling a web page form to mail responses to the operator of the page. There have been a number of vulnerabilities identified with this script. As can be seen from the table below all these alerts relate to MY.NET.150.83 and MY.NET.5.92.

Caoture Date	count(alert.src)	Source	Destination
2002-01-26 09:52:32	1	24.14.62.58	MY.NET.150.83
2002-01-27 04:46:08	1	172.167.206.93	MY.NET.150.83
2002-01-27 13:32:11	1	65.224.137.96	MY.NET.150.83
2002-01-28 03:28:42	1	65.67.113.91	MY.NET.5.92
2002-01-28 08:33:29	1	12.248.51.131	MY.NET.150.83
2002-01-29 22:53:35	1	63.21.252.14	MY.NET.150.83
2002-01-29 23:50:44	1	172.134.186.25	MY.NET.150.83
2002-01-31 10:36:52	1	165.247.11.176	MY.NET.150.83

A number of web server attacks have been directed towards both of these systems, and a L3retriever ping has also been detect originating from this address. This server should be checked for any signs of compromise. Although no alerts exist relating to trojan activity on either system.

WEB-IIS 5 Printer-beavuh 6

alert TCP $EXTERNAL any -> $INTERNAL 80 (msg: "IDS535/http -iis5-printer-beavuh";flags: P+; content: "|33 C0 B0 90 03 D8 8B 03 8B 40 60 33 DB B3 24 03 C3|";)

This alert relations to a attempt by the beavuh trojan to connect to and infect the local machine, using a vulnerability. in Internet Information Server 5 (running on Microsoft Windows 2000). All 6 alerts are relate to connections from 64.226.244.176 to MY.NET.5.79. The only alerts with MY.NET.5.79 as the source relate to a BSD type ICMP echo request to MY.NET.5.1. This would indicate that the system is running a BSD UNIX operating system, rather than MS Windows.

This trojan uses the specific string of unicode listed in the rule above, so the chances of a false positive are remote.

If this is the case then there is no need for concern relating to this alert. There is no other suspicious activity from MY.NET.5.79 that may indicate a compromise.

It is possible that a attack tool, rather than a trojan is the source of this attack. As the trojan is seeking to exploit know problems with IIS 5 it is likely that other programs also exist to take advantage of this vulnerability.

WEB-IIS encoding access 2

As with the beavuh alert above this alert relies upon identifying unicode in the URI of a http request. In this case the code is 25 31 75. The source of this alert were 216.107.3.200 and 68.33.30.61. Both of these systems were targeting MY.NET.5.96. As mentioned earlier this s ystem does not appear to be compromised.

For more details on Unicode problems with IIS see

http://www.cve.mitre.org/cgi -bin/cvename.cgi?name=CVE -2000-0884

http://www.securityfocus.com/bid/1806

WEB-MISC webdav search access 1

WebDAV (Web Distributed Authoring and Versioning) has a buffer overflow that allows a attacker to list directories in the root web directory. Microsoft has issued a patch for this vulnerability. Again the target of this attack was MY.NET.5.96. The source was 12.91.164.96

http://www.securityfocus.com/bid/2483

WEB-IIS 5 .printer isapi 1

This is a attempt to exploit a buffer overflow condition in the web printing functionality of Microsoft's IIS 5. This buffer overflow will allow the execution of code on the system. A patch has been issued by microsoft. The target of this attack was MY.NET.5.249, from 64.226.244.176. A brief analysis of alerts triggered by 64.226.244.176 show that over a period of 5 seconds 5 local hosts were scanned for IIS version 5 vulnerabilities. 6 minutes later a scan was made of 1 more host, again for a IIS 5 vulnerability. Th e exploits tested are all relatively current. This is a indication that 64.226.244.176 is focused in his attack and should be watched for further activity.

It should also be noted that the source port is the same (3770) for three of the alerts.

Capture Date	Snort Message	Source : sport	Destination : dport
2002-01-29 09:14:32	WEB-IIS 5 Printer-beavuh	64.226.244.176 : 3600	MY.NET.5.79 : 80
2002-01-29 09:14:32	WEB-IIS 5 Printer-beavuh	64.226.244.176 : 3616	MY.NET.5.95 : 80
2002-01-29 09:14:32	WEB-IIS 5 Printer-beavuh	64.226.244.176 : 3617	MY.NET.5.96 : 80
2002-01-29 09:14:34	WEB-IIS 5 .printer isapi	64.226.244.176 : 3770	MY.NET.5.249 : 80
2002-01-29 09:14:35	WEB-IIS 5 Printer-beavuh	64.226.244.176 : 3613	MY.NET.5.92 : 80
2002-01-29 09:14:37	EXPLOIT x86 NOOP	64.226.244.176 : 3770	MY.NET.5.249 : 80
2002-01-29 09:14:37	WEB-IIS 5 Printer-beavuh	64.226.244.176 : 3770	MY.NET.5.249 : 80
2002-01-29 09:22:15	WEB-IIS 5 Printer-beavuh	64.226.244.176 : 1576	MY.NET.150.83 : 80

The range in ports is a possible indication that 64.226.244.176 is running multiple scans across multiple networks. All of the destination hosts in this scan should be checked for possible compromise.

Ftp alerts

FTP DoS ftpd globbing 1013

This alert relates to a number of known problems with various ftp deamons. Vulnerable systems allow a buffer overflow if the user is able to create directories on the local system. This then enables the user to pass large amounts of data (>512 bytes) to the ftp deamon. At this point a buffer overflow occurs. This may allow the attacker to run commands on the system, or cause a denial of service. The table below shows the source and session times of these alerts.

Session Start	Session End	Source : sport	Destination : dport
2002-01-29 17:18:09	2002-01-29 17:18:17	132.238.70.240 : 2600	MY.NET.150.145 : 21
2002-01-29 17:30:02	2002-01-29 17:36:32	137.142.175.175 : 4448	MY.NET.150.145 : 21
2002-01-29 21:10:15	2002-01-29 21:12:01	64.161.36.66 : 56055	MY.NET.150.145 : 21
2002-01-30 16:47:09	2002-01-30 16:56:04	137.142.181.128 : 2201	MY.NET.153.152 : 21
2002-01-30 17:02:55	2002-01-30 17:03:18	47.82.18.193 : 1312	MY.NET.153.152 : 21
2002-01-30 19:29:04	2002-01-30 19:49:39	68.40.248.44 : 2576	MY.NET.150.145 : 21
2002-01-31 16:30:02	2002-01-31 16:31:11	141.140.108.26 : 3769	MY.NET.88.199 : 21
2002-01-31 18:05:23	2002-01-31 18:11:00	24.160.75.246 : 65168	MY.NET.150.145 : 21

The main target of this alert is MY.NET.150.145. This system do es not appear to have placed in a uncontactable state by any of these connections. 137.142.175.175 is able to connect to the server, 12 minutes after the alerts from 132.238.70.240 were generated. One disturbing record relating to MY.NET.150.145 is that it appears to be running the L3 retriever security scanning tool. This must be considered a risky situation as if the FTP server is compromised the attacker will then have access to the security tool.

FTP Exploit AIX Overflow 1

This alert detects attempt s to exploit a buffer overflow condition in the AIX ftp server. For further reference see http://www.cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-1999-0789. The attack is targete d at MY.NET.153.152. This host appears to be using a windows operating system a number of windows ICMP echo request packets have been detected originating from this host. There are also a number of other suspicious alerts relating to this system. For m ore details see the recommendations section.

MYPARTY - possible My Party infection 525

MYPARTY is a microsoft outlook email worm/virus, that also installs a trojan on a infected machine. The rule that generated this alert is no longer available so exact detection methods cannot be determined. These alerts start at 09:34:31 29/01/02 and finish at 16:58:42 on the same day. According to symantac (http://www.sarc.com/avcenter/venc/data/ w32.myparty@mm.html) this virus was only programmed to operate between the 24 th of January until the 29 th. All hosts identified by this alert should have their virus definitions updated and be scanned for virii immediately (if it has not already been don e).

min(cap_date)	max(cap_date)	count(src)	Source	:sport	Destination	:dport
2002-01-29 13:00:39	2002-01-29 13:52:59	54	MY.NET.153.142	:1202	209.151.250.170	:80
2002-01-29 10:21:46	2002-01-29 13:12:29	168	MY.NET.153.145	:1101	209.151.250.170	:80
2002-01-29 09:34:41	2002-01-29 10:32:03	41	MY.NET.153.181	:1117	209.151.250.170	:80
2002-01-29 15:14:27	2002-01-29 16:58:42	262	MY.NET.153.211	:1069	209.151.250.170	:80

Incomplete packet fragments discarded 179

This alert is not part of the current rule set. It appears to have been triggered by ICMP type 11 (this ICMP message is triggered when a router discards a packet after the ttl has reached 0). These alerts all relate to packets from external hosts to internal hosts. As more details about this alert cannot be obtained no further information can be given.

INFO alerts

INFO alerts, as the name suggest are designed to provide information about traffic found on the network. Generally this traffic is not malicious in itself, although it may provide a pointer to systems that have been compromised. One example of this is IRC traffic. Often once a machine has been compromised the attacker will install a IRC or ICQ server on the system and then 'own' the new IRC chat rooms. In this case as the network under analysis it would be a expectation that students have installed various file sharing utilities on their computers. This may or may fall within the University acceptable use policy. This is something that is up to the local administrators to deal with. Nine different INFO alerts have been generated on this system. Four of those alert types relate to the GNUTella file sharing application.

INFO Inbound GNUTella Connect request 1011
INFO Outbound GNUTella Connect request 113
INFO Inbound GNUTella Connect accept 57
INFO Outbound GNUTella Connect accept 57

These four alerts refer to the GNUTella file sharing application. This traffic is not considered malicious, u nless there is the possibility of confidential documents being leaked. This is unlikely to be the case on a university network.

The Snort rules used to detect GNUTella traffic examine the content of TCP packets for the words **GNUTELLA CONNECT** or **GNUTELLA O K.** The direction of traffic is used to determine if the request is inbound or outbound. The result of this is that the chances of a false positive are remote.

The high level of inbound connection requests indicate that a server has most likely been running on the network has now been disabled.

Only two internal hosts appear to be using GNUTella, MY.NET.153.211, MY.NET.153.203. Three other internal hosts are still receiving attempted connections from external hosts, MY.NET.151.97, MY.NET.153.157 and MY.NET.153.196.

INFO Possible IRC Access 96
Internet Relay Chat is a common chat tool, as with GNUTella IRC traffic is not generally malicious. However attackers have been know to install IRC servers on a compromised system.
The Snort rule used to genera te this alert checks for packets with a destination port between 6666 and 7000 containing the work **NICK.** The chances of a false positive are higher than with GNUTella but are still moderately low.

The top ten local sources of IRC connections are listed below

INFO FTP anonymous FTP 60
Anonymous FTP connections are only of concern if there are no anonymous FTP servers on the network. Another use for compromised systems is as a 'warz' source. These servers are used to store pirated software to be shared and traded by underground 'warez' groups.

All the detect anonymous FTP connections have are from three external sources, 211.44.115.20 (33 connections), 217.84.183.149 (14 connections) and 200.255.204.46

count(src)	Source	:sport	Destination	:dport
20	MY.NET.153.164	:2820	194.47.161.38	:6667
19	MY.NET.150.165	:1249	216.152.64.151	:6667
18	MY.NET.88.181	:2793	207.68.167.253	:6667
7	MY.NET.88.159	:1823	211.63.185.159	:6667
5	MY.NET.153.147	:1495	151.189.12.20	:6666
5	MY.NET.153.151	:1138	209.130.30.130	:6667
3	MY.NET.153.167	:1417	211.63.185.135	:6667
3	MY.NET.153.199	:1206	207.68.167.253	:6667
3	MY.NET.153.210	:1868	207.68.167.253	:6667
2	MY.NET.153.206	:1420	206.167.75.78	:6667

(13 connections). These connections were to the 22 local hosts listed below. If the network administrator is not aware of ftp servers operating on these hosts further investigation should be made.

Count (dst)	Source	:sport	Destination	:dport
4	200.225.204.46	:4712	MY.NET.150.41	:21
4	200.225.204.46	:4914	MY.NET.150.243	:21
4	200.225.204.46	:4902	MY.NET.150.231	:21
4	200.225.204.46	:4943	MY.NET.88.190	:21
4	200.225.204.46	:4940	MY.NET.88.187	:21
4	200.225.204.46	:1682	MY.NET.153.220	:21
4	200.225.204.46	:4866	MY.NET.150.195	:21
3	211.44.115.20	:4344	MY.NET.150.147	:21
3	211.44.115.20	:4280	MY.NET.150.83	:21
3	200.225.204.46	:1681	MY.NET.153.219	:21
3	211.44.115.20	:4281	MY.NET.150.84	:21
3	200.225.204.46	:4891	MY.NET.150.220	:21
3	200.225.204.46	:4868	MY.NET.150.197	:21
3	200.225.204.46	:4687	MY.NET.150.16	:21
2	200.225.204.46	:4810	MY.NET.150.139	:21
2	211.44.115.20	:4566	MY.NET.151.114	:21
2	200.225.204.46	:4897	MY.NET.150.226	:21
1	211.44.115.20	:4824	MY.NET.5.95	:21
1	217.84.183.149	:4684	MY.NET.150.107	:21
1	217.84.183.149	:3580	MY.NET.5.137	:21
1	211.44.115.20	:4814	MY.NET.5.85	:21
1	211.44.115.20	:4821	MY.NET.5.92	:21

Table 6Anonymous FTP connections

INFO Napster Client Data 26

Napster is a peer to peer file sharing program similar to GNUTella (in fact GNUTella is based upon Napster). In this case all alerts are from a single local source MY.NET.153.171. To 22 different external hosts.

INFO - ICQ Access 13

ICQ is a online chat application. It can also be used for sending control information to trojans. All 13 ICQ alerts are sourced from a single host on the internal network, MY.NET.5.238. This host has established ICQ connections with three external hosts, over a unusual time frame and for s hort periods of time. The alert rule for ICQ should detect all ICQ traffic. In this case the low level of ICQ traffic does not fit in with the pattern of traffic of a standard ICQ user. A list of all ICQ alerts is shown below

Capture Time	Source : sport	Destination : dport
2002-01-28 15:25:01	MY.NET.5.238 : 2614	64.12.164.193 : 80
2002-01-28 15:25:04	MY.NET.5.238 : 2616	205.188.250.25 : 80
2002-01-28 15:25:30	MY.NET.5.238 : 2619	205.188.248.57 : 80
2002-01-28 15:25:41	MY.NET.5.238 : 2623	205.188.248.57 : 80
2002-01-28 15:25:49	MY.NET.5.238 : 2624	205.188.250.25 : 80
2002-01-28 15:26:03	MY.NET.5.238 : 2625	205.188.250.25 : 80
2002-01-28 15:26:17	MY.NET.5.238 : 2626	205.188.250.25 : 80
2002-01-28 15:26:59	MY.NET.5.238 : 2627	205.188.248.57 : 80
2002-01-29 16:07:10	MY.NET.5.238 : 2689	205.188.250.25 : 80
2002-01-29 16:07:11	MY.NET.5.238 : 2691	205.188.250.25 : 80
2002-01-29 16:07:12	MY.NET.5.238 : 2692	205.188.250.25 : 80
2002-01-29 16:07:12	MY.NET.5.238 : 2693	205.188.250.25 : 80
2002-01-29 16:07:19	MY.NET.5.238 : 2698	205.188.250.25 : 80

For two days in a row this host connects via ICQ to three different external hosts. With a destination port of 80, a port assigned for use with http. The period of connection is for only 2 minutes on the 28 th and 9 seconds on the 29 th. This is not enough time to carry on a conve rsation. Further more the range of source ports indicates 4 other network clients are started during the time the ICQ sessions are running on the 28 th and 6 other network clients are started in the 9sec the ICQ session runs for on the 29 th.

The only other alert relating to MY.NET.5.238 is a NIMDA alert indicating that this machine has attempted to execute cmd on a remote host. If this machine was infected with the NIMDA virus we would expect to see a large number of attempts to infect other hosts, rather than a single attempt. It is unlikely that this machine is infected with the Nimda virus. It is possible that this s machine has been compromised and is now being used in a attempt to compromise other systems.

INFO - Possible Squid Scan 11

Squid is a p opular proxy server. Attackers commonly use proxy servers as a method of hiding there identification from potential victims. This snort rule detects TCP SYN packets destined for port 3128 on the local network. Even if this traffic is not a squid scan it can still be considered suspicious. Of the eleven alerts only the four connections from 66.45.42.38 appear to be the result of a missconfiguration. In this

Capture Date	Source : sport	Destination : dport
2002-01-26 02:20:45	216.152.64.163 : 36286	MY.NET.150.165 : 3128
2002-01-28 10:11:05	24.162.192.226 : 1520	MY.NET.153.164 : 3128
2002-01-30 14:06:18	204.152.186.58 : 2831	MY.NET.152.178 : 3128
2002-01-30 22:12:44	24.102.145.17 : 4599	MY.NET.153.140 : 3128
2002-01-30 22:50:58	66.45.42.38 : 3891	MY.NET.153.198 : 3128
2002-01-30 22:50:59	66.45.42.38 : 3891	MY.NET.153.198 : 3128
2002-01-30 22:50:59	66.45.42.38 : 3891	MY.NET.153.198 : 3128
2002-01-30 22:51:00	66.45.42.38 : 3891	MY.NET.153.198 : 3128
2002-01-30 23:24:08	216.152.64.163 : 59316	MY.NET.153.162 : 3128
2002-01-31 12:04:55	204.152.186.58 : 2073	MY.NET.152.180 : 3128
2002-01-31 20:25:38	204.152.186.58 : 3678	MY.NET.152.180 : 3128

case the source clearly makes four attempts to connect to the squid server.

It is recommended t hat the network administrator confirm if any of the local hosts in

this alert are in fact squid servers. Squid is sometimes installed by default or mistake on some open source operating systems. Access to port 3128 should be blocked to external sources.

MISC Alerts

MISC traceroute 117

Traceroute is a technique used to identify routers and other network devices between two hosts. The Snort rule used for this alert is no longer part of the Snort ruleset, so analysing this alert is slightly more challengi ng. A number of attempts have been made to connect to port 1214. There has been a high level of traffic reported to this port on dshield (http://www.dshield.org/port_report.php?port=1214). A common use for this port is the Kazaa file sharing application.

count(src)	source	:sport	destination	:dport
26	192.204.106.2	:36558	MY.NET.150.41	:1214
20	192.168.0.3	:1860	MY.NET.150.133	:1214
13	203.201.28.102	:1180	MY.NET.88.162	:1214
13	194.64.241.6	:6921	MY.NET.150.41	:1214
12	203.201.22.239	:1241	MY.NET.150.41	:1214
8	203.201.28.11	:1135	MY.NET.88.162	:1214
4	203.201.20.244	:1382	MY.NET.150.41	:1214
4	163.151.0.253	:25304	MY.NET.88.162	:1214
4	199.213.179.188	:1223	MY.NET.150.133	:1214
3	62.220.192.4	:1482	MY.NET.150.41	:1214
3	208.240.88.84	:52755	MY.NET.5.92	:33474
3	199.213.180.77	:4482	MY.NET.150.133	:1214
2	195.116.34.118	:34372	MY.NET.150.133	:33498
1	64.49.141.122	:2517	MY.NET.88.162	:1214
1	157.238.46.11	:2138	MY.NET.153.206	:33502

As traceroute is not malicious there is little further information to be gained from this alert. However this information may be used to correlate other activity identified by other alerts.

MISC Large ICMP Packet 42

ICMP packets are not usually more than 512bytes, although ICMP packets may be larger than this for legitimate reasons. Large ICMP packets are also commonly used in Denial of Service attacks, although during a DoS attack a large number of large ICMP packets would be seen. Sixteen different connections have generated the large ICMP alert. This information is not a indication of a attack, but may provide insight to a attack when correlated with other alerts.

MISC source port 53 to <1024 8

This alert is triggered by a TCP SYN packet from source port 53 (DNS) to another port below 1024. The only time this type of traffic should be seen is between DNS

servers conducting a zone transfer. This should only occur between local DNS systems. The eight alerts generated by this rule all are from external hosts attempting to connect to internal systems.

count(src)	Source	:sport	Destination	:dport
1	134.75.30.1	:53	MY.NET.152.137	:53
1	192.33.4.12	:53	MY.NET.152.137	:53
1	198.41.0.10	:53	MY.NET.152.137	:53
1	202.12.28.131	:53	MY.NET.152.137	:53
1	210.180.98.69	:53	MY.NET.152.137	:53
1	210.94.0.7	:53	MY.NET.152.137	:53
1	216.106.173.147	:53	MY.NET.152.181	:256
1	63.146.181.107	:53	MY.NET.88.155	:212

Six of the detected packets are to MY.NET.152.137. The only other alerts relating to this system are *ICMP address mask reply* to local hosts. This is unusual as normally ICMP *address mask rely* would be generated by a router in response to a request from the client. It is unlikely that a router is also acting as a DNS server. As this alert is triggered by a TCP SYN packet there is no confirmation of a successful connection. It is possible that the external hosts are scanning in a search for DNS servers, (although this is also unlikely as a whois lookup will provide a list of the registered DNS servers for the university). A search for other alerts relating to the source of these packets shows that no other alerts relating to these external sources have been raised.

There is a large amount of suspicious activity relating to MY.NET.152.181. This has been mentioned in the Red Worm section, and will be covered in more detail in the recommendations section on compromised systems.

The local host MY.NET.88.155 is the destination of a large amount of activity but does not appear to be generating any itself. It does not appear to be compromised.

MISC PCAnywhere Startup 1

PCAnywhere is a remote control application that allows a user to connect to their computer over the internet, or through a modem dial up connection. In this case it appears that the remote host 216.150.152.145 is connecting to the local host MY.NET.5.92. Although PCAnywhere does provide a certain level of authentication and access control it is not recommended to allow remote access of this type from external systems.

There are four bugtraq entries referring to problems with PCAnywhere

25-04-2000: Symantec pcAnywhere Port Scan DoS Vulnerability
10-04-2000: PCAnywhere Denial of Service Vulnerability
06-04-2000: Symantec pcAnywhere Weak Encryption Vulnerability
11-05-1999: PCAnywhere32 Denial of Service Vulnerability

The local host is the source of a number *web server 403 access forbidden alerts*. This would indicate that this host is running a web server. It is strongly recommended that the PCAnywhere application be removed, or the local ports are restricted to only allow connections to PCAnywhere from a specific external host.

Exploit alerts

EXPLOIT NTPDX buffer overflow 179

This is an attempt to exploit a buffer overflow condition in the Network Time Protocol service. If this buffer overflow is successful the attacker may be able to gain administrator or root access to the system. For more details see: http://online.securityfocus.com/archive/1/174011 Both windows and UNIX systems running the NTP service appear to be vulnerable t o this attack. Eighteen local system were targeted by this attack. All of these systems should be checked to ensure that the appropriate patches have been applied.

The four external hosts using the port 123 are most probably using legitimate NTP information exchange.

count(src)	Source	:sport	Destination	:dport
36	64.152.108.141	:123	MY.NET.88.163	:123
32	63.210.134.141	:1853	MY.NET.88.163	:123
31	66.38.185.143	:1488	MY.NET.88.163	:123
21	64.152.108.142	:1803	MY.NET.88.163	:123
13	216.106.173.149	:1059	MY.NET.153.210	:123
11	211.106.66.159	:1094	MY.NET.153.113	:123
9	216.106.172.148	:2420	MY.NET.153.210	:123
5	216.106.172.157	:123	MY.NET.152.181	:123
5	216.106.172.156	:1388	MY.NET.152.181	:123
4	66.38.171.142	:1062	MY.NET.88.163	:123
3	66.77.13.104	:2420	MY.NET.88.155	:123
2	140.142.8.72	:3853	MY.NET.153.157	:123
2	63.146.181.103	:1067	MY.NET.88.155	:123
1	63.146.181.107	:1176	MY.NET.88.155	:123
1	66.77.13.103	:1055	MY.NET.88.155	:123
1	63.146.181.116	:123	MY.NET.88.155	:123
1	216.106.172.149	:2074	MY.NET.153.210	:123
1	211.106.66.157	:123	MY.NET.153.106	:123

A list of alerts generated by hosts targeted by the NTPDX buffer overflow reveals that MY.NET.153.210 and MY.NET.153.157 are sources of http unicode alerts. Both of these systems should be examine for possible compromise.

EXPLOIT x86 NOOP 35

Unfortunately the rule relating to this alert cannot be found in the SNORT rule database. By examining the information available in the alerts it appears this rule as set to detect TCP packets to the local network with specific options set. The destination ports are not consistent, although the source ports (especially port 80) imply that the majority of these alerts are false positives.

count(src)	Source	:sport	Destination	:dport
12	68.55.200.56	:445	MY.NET.150.143	:1180
6	128.205.32.51	:35185	MY.NET.153.195	:1414
4	205.138.230.234	:80	MY.NET.153.202	:2655
3	207.46.177.148	:80	MY.NET.152.19	:1848
1	216.34.218.99	:80	MY.NET.153.146	:3415
1	211.192.139.39	:80	MY.NET.153.184	:1319
1	142.177.200.180	:1419	MY.NET.150.49	:1214
1	64.226.244.176	:3770	MY.NET.5.249	:80
1	64.152.108.141	:0	MY.NET.88.163	:0
1	209.1.225.217	:80	MY.NET.152.215	:3781
1	202.108.36.49	:80	MY.NET.153.118	:2175
1	152.20.1.24	:80	MY.NET.88.191	:1624
1	206.201.228.250	:80	MY.NET.153.142	:3362
1	204.71.201.134	:80	MY.NET.153.208	:2184

Three of the hosts (MY.NET.153.202, MY.NET.153.184 and MY.NET.153.146) targeted by this attack were also the sour ce of http IIS unicode alerts. As mentioned in that section the unicode alerts may be false positives. However it is likely that these systems have been used to launch attacks against external hosts. These systems should be investigated.

EXPLOIT x86 set gid 0 6

This alert detects any TCP packets containing the hex sequence b0b5 cd80 this is used to change group permissions on a UNIX system. This alert has a high chance of false positives as it is only seeking a match on 4 bytes of data. The table below shows the source and destinations of this alert. The traffic with a source port of 80 is most likely harmless http replies. The three other packets may be a source of attack.

count(src)	Source	:sport	Destination	:dport
1	165.123.150.170	:1214	MY.NET.150.41	:1342
1	216.131.68.2	:80	MY.NET.152.169	:2017
1	24.138.59.130	:6968	MY.NET.152.21	:3821
1	129.64.99.132	:80	MY.NET.153.144	:2321
1	211.192.192.22	:1755	MY.NET.153.153	:4438
1	202.101.10.208	:80	MY.NET.153.187	:1954

Of these destination hosts three (MY.NET.153.144, MY.NET.153.153 and MY.NET.153.187) are also the source of IIS unicode alerts.

EXPLOIT x86 setuid 0 8

As with the previous alert this rule attempts to detect remote users changing permissions on files. In this case the rule is search for the hex sequence b017 cd80. The likely hood of false positives is also high for this alert.

count(src)	Source	:sport	Destination	:dport
1	63.208.2.93	:80	MY.NET.150.176	:2138
2	130.64.136.239	:1214	MY.NET.150.49	:1867
1	64.152.216.37	:1755	MY.NET.151.95	:1789
1	61.136.61.21	:80	MY.NET.153.185	:4508
1	61.136.61.21	:80	MY.NET.153.187	:1774
1	61.138.3.107	:20518	MY.NET.153.45	:6970
1	202.108.36.211	:80	MY.NET.88.165	:1582

The local hosts MY.NET.153.185, MY.NET.153.187 and MY.NET.88.165 were also the source of *IIS unicode* alerts.

EXPLOIT x86 stealth noop 2

As with the previous alert this rule attempts to detect remote users changing permissions on files. In this case the rule is search for the hex sequence eb 02 eb 02 eb 02. The likely hood of false positives is also high for this alert.

count(src)	Source	:sport	Destination	:dport
1	216.26.144.52	:80	MY.NET.153.118	:1749
1	209.249.147.177	:80	MY.NET.153.184	:1556

As mentioned earlier MY.NET.152.184 has been the source of *IIS unicode* alerts against external hosts.

TFTP - Internal TCP connection to external tftp server 113
TFTP - External UDP connection to internal tftp server 6
TFTP - Internal UDP connection to external tftp server 2

TFTP (trivial file transfer protocol) is commonly used by routers to update configuration. It is extremely unusual to allow TFTP to leave the internal network. As with the previous exploit alerts the TFTP rules are not listed in the current Snort rules database. Moreover these rules appear to be misconfigured. A list of the TFTP - Internal TCP connection to external tftp server reveals 64 alerts with a source of 142.176.138.126 and destination of MY.NET.88.181. This does not appear to match the message associated with the rule. There are three local systems associated with these alerts, MY.NET.88.181, MY.NET.88.163 and MY.NET.88.155. These hosts are not listed in any other alerts. It is possible these are routers and they have been compromised. If they are routers they should be examined for possible problems.

SCANS

A large amount of scanning has been detected, both entering and leaving the local network. As mentioned at the beginning of this report scans are not malicious, they are a information gathering exercise. In some cases scans may provide a warning that a attacker is targeting the local system, and the vulnerabilities they are looking for.

The table below shows the most common UDP and TCP ports scanned on the local network. A list of the top 10 scanners from the internal and external networks is also provided. Scanning is generally considered to be background noise to the traffic on the internet. However it is considered to be impolite to scan other peoples networks. Local hosts that are a source of excessive scanning should be investigated and their users notified of the university acceptable use policy.

It is also common for a attacker to use a compromised system to perform large scans of other systems in a search for further systems to compromise. There do not appear to be any local systems generating this type of scan.

SCAN Proxy attempt 111

SCAN Synscan Portscan ID 19104 93

SCAN FIN 2

Queso fingerprint 11

Queso is a tool used to identify the operating system of remote host.

Russia Dynamo - SANS Flash 28 -jul-00 16

This alert is no longer listed in the Snort rules database. As the message implies this alert appears to relate to a alert made by SANS in July 2000. A search of the SANS website (www.sans.org) fails to reveal any reference to Russia dynamo. This alert appears to be trigger by traff ic to and from port 1214 and possibly the class B network 194.87.0.0. TCP port 1214 is used by the Morpheus and Kazaa file sharing applications. (http://users.pandora.be/lechat/Morpheus%20 Exploit.htm). There are no other alerts relating to this address. This alert is probably safe to ignore.

Capture date	Source	: sport	Destination	:dport
2002-01-28 23:06:12	194.87.6.142	: 2146	MY.NET.88.162	:1214
2002-01-28 23:06:12	MY.NET.88.162	: 1214	194.87.6.142	:2146
2002-01-28 23:06:15	194.87.6.142	: 2146	MY.NET.88.162	:1214
2002-01-28 23:06:15	194.87.6.142	: 2146	MY.NET.88.162	:1214
2002-01-28 23:06:15	MY.NET.88.162	: 1214	194.87.6.142	:2146
2002-01-28 23:06:17	194.87.6.142	: 2146	MY.NET.88.162	:1214
2002-01-28 23:06:17	MY.NET.88.162	: 1214	194.87.6.142	:2146
2002-01-28 23:06:17	MY.NET.88.162	: 1214	194.87.6.142	:2146
2002-01-28 23:06:17	MY.NET.88.162	: 1214	194.87.6.142	:2146
2002-01-28 23:06:18	194.87.6.142	: 2146	MY.NET.88.162	:1214
2002-01-28 23:06:18	MY.NET.88.162	: 1214	194.87.6.142	:2146
2002-01-28 23:06:18	194.87.6.142	: 2146	MY.NET.88.162	:1214
2002-01-28 23:06:19	194.87.6.142	: 2146	MY.NET.88.162	:1214
2002-01-28 23:06:19	MY.NET.88.162	: 1214	194.87.6.142	:2146
2002-01-28 23:06:19	194.87.6.142	: 2146	MY.NET.88.162	:1214
2002-01-28 23:06:20	194.87.6.142	: 2146	MY.NET.88.162	:1214

Back Oriface 13

Back Oriface is a trojan program developed by Cult of the Dead Cow. (http://www.cultdea dcow.com). This program allows remote users to take control of the infected windows host. In the original version of the trojan the port was programmed to 31337. In later versions it was modified so that the user could configure the port. This rule ap pears to have been configured to detect attempted connections to port 31337. Current rules for back oriface examine the content of the packets for more information. The rule used in this case has a high possibility of false positives. Although in this c ase it appears that all attempts are genuine efforts to connect to a back oriface server. The most disturbing aspect of these alerts are that both the source and destination are from internal addresses. This may indicate that local systems have been comp romised by a eternal user, or local users are attempting to compromise local systems. In either case all hosts mentioned in this alert should be investigated further.

cap_date	Source	: sport	Destination	: dport
2002-01-27 15:22:37	216.106.172.149	: 25220	MY.NET.152.181	: 31337
2002-01-28 13:20:17	MY.NET.6.49	: 27492	MY.NET.153.197	: 31337
2002-01-28 14:34:34	MY.NET.6.48	: 12554	MY.NET.153.161	: 31337
2002-01-28 15:34:53	MY.NET.6.52	: 29281	MY.NET.153.166	: 31337
2002-01-28 16:05:24	MY.NET.6.52	: 12846	MY.NET.153.208	: 31337
2002-01-28 17:13:51	MY.NET.6.48	: 25193	MY.NET.153.184	: 31337
2002-01-28 17:14:30	MY.NET.6.48	: 25193	MY.NET.153.184	: 31337
2002-01-28 17:22:44	MY.NET.6.49	: 43945	MY.NET.153.193	: 31337
2002-01-30 19:06:03	MY.NET.6.49	: 44203	MY.NET.153.179	: 31337
2002-01-31 09:19:48	MY.NET.6.52	: 21365	MY.NET.153.179	: 31337
2002-01-31 09:19:48	MY.NET.6.52	: 21365	MY.NET.153.179	: 31337
2002-01-31 10:49:25	MY.NET.6.49	: 30313	MY.NET.153.195	: 31337
2002-01-31 10:49:25	MY.NET.6.49	: 30313	MY.NET.153.195	: 31337

TCP src and dst outside network 11

This alert indicates packets detected inside the lo cal network with external source and address. This should not occur unless a local system is misconfigured.

cap_date	Source	:sport	Destination	:dport
2002-01-25 12:08:50	172.140.78.51	:80	210.101.143.52	:4046
2002-01-25 13:49:17	172.129.64.168	:80	198.68.179.2	:34478
2002-01-25 14:29:52	172.129.64.168	:1243	12.232.40.117	:1271
2002-01-25 14:35:07	172.129.64.168	:80	24.28.198.150	:3296
2002-01-25 14:35:11	172.129.64.168	:80	24.28.198.150	:3296
2002-01-25 16:22:49	172.140.212.25	:80	66.25.166.236	:2444
2002-01-25 16:22:52	172.140.212.25	:80	66.25.166.236	:2444
2002-01-25 16:34:46	172.140.212.25	:27374	213.1.138.142	:1025
2002-01-25 16:34:50	172.140.212.25	:27374	213.1.138.142	:1025
2002-01-25 16:58:48	172.140.212.25	:80	208.26.170.125	:3411

It should be noted that some of these alerts appear twice, this could be caused by two different Snort sensors detecting the same packets, how ever the capture time is exactly identical. It is more likely that one sensor have written these alerts to the log file twice. The configuration of this sensor should be checked.

The port 27374 is commonly used by the subseven trojan. It is possible t his host has been compromised. It is also possible a local host is spoofing this source address.

All source address are from AOL.

America Online, Inc. (NETBLK -AOL-172BLK)
12100 Sunrise Valley Drive
Reston, VA 20191
US
Netname: AOL -172BLK
Netblock: 172.128.0.0 - 172.191.255.255
Maintainer: AOL
Coordinator:
America Online, Inc. (AOL -NOC-ARIN) domains@AOL.NET
703-265-4670
Domain System inverse mapping provided by:

DAHA-01.NS.AOL.COM 152.163.159.233
DAHA-02.NS.AOL.COM 205.188.157.233
ADDRESSES WITHIN THIS BLOCK ARE NON -PORTABLE
Record last updated on 28 -Mar-2001.
Database last updated on 15 -Mar-2002 19:57:41 EDT.
The ARIN Registration Services Host contains ONLY Internet
Network Information: Networks, ASN 's, and related POC's.
Please use the whois server at rs.internic.net for DOMAIN related
Information and whois.nic.mil for NIPRNET Information.

Tiny Fragments - Possible hostile activity

Fragmentation was a technique commonly used to evade NIDS. However now most NIDS specifically check for small fragments. In this case 68.36.73.138 has sent four fragmented packets to MY.NET.153.211 on two separate occasions over extremely short time period (under 1 second). There is no standard reason for this.

sec_id	Source	sport	Destination	dport
20020128195741.8712910	218.75.164.147	0	MY.NET.153.185	0
20020131102809.7661150	68.36.73.138	0	MY.NET.153.211	0
20020131102809.8636350	68.36.73.138	0	MY.NET.153.211	0
20020131102809.8848810	68.36.73.138	0	MY.NET.153.211	0
20020131102816.7553250	68.36.73.138	0	MY.NET.153.211	0
20020131102816.8492550	68.36.73.138	0	MY.NET.153.211	0
20020131102816.8534390	68.36.73.138	0	MY.NET.153.211	0
20020131102816.9037620	68.36.73.138	0	MY.NET.153.211	0

A review of other traffic from 68.36.7 3.138 reveals that the only alerts from this source are the tiny fragments. The local host MY.NET.153.211 has already been identified as being possibly infected with the My Party virus and the source of *IIS unicode* alerts. The probability of this machine being compromised has just increased.

SMB CD..

This alert refers to attempts by hosts using the SMB protocol to share files to access the file system one directory level above the current shared drive they are using. This alert is based upon packets of the SMB protocol containing the characters CD.. The chances for false positives are fairly low.

cap_date	Source	: sport	Destination	: dport
2002-01-29 10:27:07	MY.NET.5.247	: 1266	MY.NET.5.141	: 139
2002-01-29 10:27:07	MY.NET.5.247	: 1266	MY.NET.5.141	: 139
2002-01-29 10:27:07	MY.NET.5.247	: 1266	MY.NET.5.141	: 139
2002-01-29 10:27:07	MY.NET.5.247	: 1266	MY.NET.5.141	: 139
2002-01-29 10:27:07	MY.NET.5.247	: 1266	MY.NET.5.141	: 139
2002-01-29 10:27:07	MY.NET.5.247	: 1266	MY.NET.5.141	: 139
2002-01-29 10:27:07	MY.NET.5.247	: 1266	MY.NET.5.141	: 139
2002-01-29 11:01:49	MY.NET.5.141	: 1126	MY.NET.70.156	: 139

Seven of these alerts are generated by one attempt by MY.NET.5.247 to access MY.NET.5.141.

This alert is of no concern.

Port 55850 UDP - Possible myserver activity - ref. 010313-1 6

My server is another trojan, which commonly uses UDP port 55850. The traffic associated with this alert is all suspicious, all local hosts should be investigated.

cap_date	Source	:	sport	Destination	:	dport
2002-01-30 09:02:58	MY.NET.6.48	:	13936	MY.NET.153.154	:	55850
2002-01-30 12:19:24	66.77.13.105	:	55850	MY.NET.88.155	:	50997
2002-01-30 12:25:50	63.210.134.141	:	55850	MY.NET.88.163	:	4292
2002-01-30 12:25:50	63.210.134.141	:	55850	MY.NET.88.163	:	4292
2002-01-30 12:25:51	63.210.134.141	:	55850	MY.NET.88.163	:	4292
2002-01-31 15:34:08	MY.NET.6.52	:	43457	MY.NET.153.148	:	55850

The connection from 63.210.134.141 to MY.NET.88.163 appears to indicate that MY.NET.88.163 is controlling the external host. This should be investigated.

SUNRPC highport access!

Remote Procedure Calls (RPC) are used to run commands on remote systems. There are a number of vulnerabilities in the RPC suite. A few are listed below:

CVE-1999-0003	Execute commands as root via buffer overflow in Tooltalk database server (rpc.ttdbserverd)
CVE-1999-0008	Buffer overflow in NIS+, in Sun's rpc.nisd program
CVE-1999-0208	rpc.ypupdated (NIS) allows remote users to execute arbitrary commands.
CVE-1999-0212	Solaris rpc.mountd generates error messages that allow a remote attacker to determine what files are on the server.
CVE-1999-0228	Denial of service in RPCSS.EXE program (RPC Locator) in Windows NT.
CVE-1999-0320	SunOS rpc.cmsd allows attackers to obtain root access by overwriting arbitrary files.
CVE-1999-0353	rpc.pcnfsd in HP gives remote root access by changing the permissions on the main printer spool directory.
CVE-1999-0493	rpc.statd allows remote attackers to forward RPC calls to the local operating system via the SM_MON and SM_NOTIFY commands, which in turn could be used to remotely exploit other bugs such as in automountd.
CVE-1999-0687	The ToolTalk ttsession daemon uses weak RPC authentication, which allows a remote attacker to execute commands.
CVE-1999-0696	Buffer overflow in CDE Calendar Manager Service Daemon (rpc.cmsd)
CVE-1999-0900	Buffer overflow in rpc.yppasswdd allows a local user to gain privileges via MD5 hash generation.
CVE-1999-0969	The Windows NT RPC service allows remote attackers to conduct a denial of service using spoofed malformed RPC packets which generate an error message that is sent to the spoofed host, potentially setting up a loop, aka Snork.
CVE-1999-0974	Buffer overflow in Solaris snoop allows remote attackers to gain root privileges via GETQUOTA requests to the rpc.rquotad service.
CVE-1999-1127	Windows NT 4.0 does not properly shut down invalid named pipe RPC connections, which allows remote attackers to cause a denial of service (resource exhaustion) via a series of connections containing malformed data, aka the "Named Pipes Over RPC" vulnerability.
CVE-1999-1258	rpc.pwdauthd in SunOS 4.1.1 and earlier does not properly prevent remote access to the daemon, which allows remote attackers to obtain sensitive system information.
CVE-2000-0508	rpc.lockd in Red Hat Linux 6.1 and 6.2 allows remote attackers to cause a denial of service via a malformed request.
CVE-2000-0771	Microsoft Windows 2000 allows local users to cause a denial of service by

	corrupting the local security policy via malformed RPC traffic, aka the "Local Security Policy Corruption" vulnerability.
CVE-2001-0331	Buffer overflow in Embedded Support Partner (ESP) daemon (rpc.espd) in IRIX 6.5.8 and earlier allows remote attackers to execute arbitrary commands.
CVE-2001-0662	RPC endpoint mapper in Windows NT 4.0 allows remote attackers to cause a denial of service (loss of RPC services) via a malformed request.
CVE-2001-0717	Format string vulnerability in ToolTalk database server rpc.ttdbserverd allows remote attackers to execute arbitrary commands via format string specifiers that are passed to the syslog function.
CVE-2001-0779	Buffer overflow in rpc.yppasswdd (yppasswd server) in Solaris 2.6, 7 and 8 allows remote attackers to gain root access via a long username.

In this case it appears the alerts are d etecting regular traffic. The fact the same two local hosts are communicating at the same time over two days makes it likely a scheduled job is being run.

cap_date	Source	:sport	Destination	:dport
2002-01-26 01:50:00	MY.NET.149.95	:947	MY.NET.6.39	:32771
2002-01-28 01:50:00	MY.NET.149.96	:937	MY.NET.6.39	:32771
2002-01-28 01:50:00	MY.NET.149.96	:937	MY.NET.6.39	:32771
2002-01-28 01:50:00	MY.NET.149.96	:937	MY.NET.6.39	:32771
2002-01-28 01:50:00	MY.NET.149.96	:937	MY.NET.6.39	:32771
2002-01-28 01:50:00	MY.NET.149.96	:937	MY.NET.6.39	:32771

RFB - Possible WinVNC - 010708-1

WinVNC is a remote control tool that allows a user to display a copy of a remote desktop of a local machine. There are known vulnerabilities with this application.

CVE-2000-1164	WinVNC installs the WinVNC3 registry key with permissions that give Special Access (read and modify) to the Everybody group, which allows users to read and modify sensitive information such as passwords and gain access to the system.
CAN-2001-0167	** CANDIDATE (under review) ** Buffer overflow in AT&T WinVNC (Virtual Network Computing) client 3.3.3r7 and earlier allows remote attackers to execute arbitrary commands via a long rfbConnFailed packet with a long reason string.
CAN-2001-0168	** CANDIDATE (under review) ** Buffer overflow in AT&T WinVNC (Virtual Network Computing) server 3.3.3r7 and earlier allows remote attackers to execute arbitrary commands via a long HTTP GET request when the DebugLevel registry key is greater than 0.

cap_date	Source	: sport	Destination	: dport
2002-01-31 13:11:18	66.200.114.147	: 5900	MY.NET.152.11	: 1941
2002-01-31 13:17:24	66.200.114.148	: 5900	MY.NET.152.11	: 1942
2002-01-31 13:17:53	66.200.114.146	: 5900	MY.NET.152.11	: 1943
2002-01-31 15:10:38	66.200.114.147	: 5900	MY.NET.152.11	: 3366

The alerts generated by this rule appear to indicate that winvnc is running on external hosts. A local host is access these remote systems.

BACKDOOR NetMetro File List

NetMetro is another trojan program. It uses ports 5031 and 5032. The snort rule used to detect the trojan in this case is looking for TCP traffic on port 5032, containing the hex values 2d 2d. In this case the alert appears to be a false positive as the internal connection is from port 80 on MY.NET.5.96, a system identified earlier as a web server. This is most likely http traffic.

2002-01-28 21:19:56 MY.NET.5.96:80 141.157.96.84:5032

Out of Scope Traffic

There was a extremely small amount of out of scope traffic detected on the network. A total of 9 OOS packets were detected over the seven day period. Most of these packets were destined for port 1214, commonly used by the KAZAA file sharing application. There was no traffic of concern in these logs.

Systems for further investigation

A number of hosts on internal network appear to be compromised. The majority of these systems are located in the MY.NET.153.0/24 subnet. This network appears to be used by students as there is also a high level of chat and file sharing activity from this address range.

The following systems have all been related with alerts indicating the systems have been compromised, that they are being used to attack other systems. In either case these systems should be check ed for problems.

MY.NET.5.238

This system has raised a number of alerts relating to ICQ traffic, this if often used by trojans to send control information. The majority of the ICQ traffic is to 205.188.250.25. A check of the registration information for this address reveals that its DNS name is cb.icq.com, and the IP address is owned by AOL. The ICQ traffic is probably legitimate.

IP Address:205.188.250.25 HostName:cb.icq.com DShield Profile:
Country: US
Contact E-mail: tosgeneral@aol.com
Total Records against IP: 285
Number of targets: 90
Date Range: 2002-03-25 to 2002-03-25
Ports Attacked (up to 10):
Port Attacks
1080 1
Whois:

America Online, Inc (NETBLK-AOL-DTC)
22080 Pacific Blvd
Sterling, VA 20166 US
Netname: AOL-DTC
Netblock: 205.188.0.0 - 205.188.255.255
Coordinator:
America Online, Inc. (AOL-NOC-ARIN) domains@AOL.NET
703-265-4670
Domain System inverse mapping provided by:
DNS-01.NS.AOL.COM 152.163.159.232
DNS-02.NS.AOL.COM 205.188.157.232
Record last updated on 27-Apr-1998.
Database last updated on 12-Mar-2002 19:58:03 EDT.

This host is also the source of alerts indicating NIMDA traffic. In all cases this traffic is to external hosts. This should be investigated further.

MY.NET.152.18

There has been a large number of NMAP or HPing requests and L3 Retriever requests from this hosts. The L3 Retriever pings may be legitimate, although the number of these packets detected, from the range of different sources does not appear to conform to the manner in which the application would normally be implemented. This system is also the destination of a number of high port packets from MY.NET.6.52. This may be the adore or red Linux worm. If this host is using a Linux operating system it should be investiga ted.

MY.NET.153.146, MY.NET.153.152, MY.NET.153.157, MY.NET.153.184, MY.NET.153.185, MY.NET.153.187, MY.NET.153.202, MY.NET.153.210

These hosts are the source of a large number of HTTP Unicode and cgi null byte alerts targeted at various external systems. Some of these systems are also the source of Red worm alerts.

MY.NET.153.142, MY.NET.153.145, MY.NET.153.181, MY.NET.153.211

These four systems appear to be infected with the Myparty virus. They should be scanned with anti virus software immediately

MY.NET.5.141, MY.NET.5.249, MY.NET.5.95, MY.NET.5.96, MY.NET.5.97, MY.NET.88.190

These systems appear to be web servers. A number of different attacks have been targeted at these servers. There is also indication that these systems are using SNMP. There are a number of known vulnerabilities with SNMP especially if it is used with a public community string. For more details see the alert section on SNMP public. It is recommended that SNMP be removed from these systems, or updated patches from the vendor should be applied.

MY.NET.150.198

This host appears to be providing print services to a large number of systems. Verify that this system does have the latest patches applied to it.

Recommendations:

In its current state the university network appears to be largely open to the outside world. It is strongly recommended that the needs of users be examined and wherever possible filtering on the border routers should be applied. In particular access to TCP & UDP ports 137, 138 and 139 should be blocked to all external sources. These ports are used by microsoft networking and file sharing. Access to these ports will provide a intruder with a substantial amount of information about the network, and may even allow a intruder to access files on local hosts.

The wide use of the SNMP protocol, without a configured community string should also be avoided. Currently there appear to be a number of different hosts using this protocol. There have recently been a large number of vulnerabilities identified with this protocol, that may allow a attacker complete access to your network. See the link provided in the alerts section relating to the SNMP alerts. Filtering should also be applied at the border router to prevent SNMP from leaving or entering the network.

A number of systems appear to be using the L3 retriever security auditing tool. However the manner in which this is implemented does not appear to conform to the recommendations of the vendor. If this tool is in use it's implementation should be checked. If the network administrator was not aware of the use of this tool all hosts detected using it should be thoroughly checked.

The use of PCAnywhere on the internal network is a point of some concern. This may allow a attacker access to the local host, and from there other hosts on the network. The use of this type of software should be covered by your computer acceptable use policy.

The trivial file transfer protocol should not be permitted to leave the local network. Again this is filtering that should b e performed by the border routers.

There appear to be a number of Snort sensors logging to a central file. This is a good method for managing intrusion detection on the network. However it would be of great assistance when analysing the log files if each alert indicated the sensor that generated it. The Snort rule sets in use also appear to be substantially out of date. A number of alerts, relate to warnings that have now expired, and it is likely that new rules have been developed to detect new vulnera bilities.

One of the Snort systems appears to be making multiple entries in the log files. The configuration of this system should also be investigated.

Method of analysis

In order to facilitate the analysis process the data from the log files were l oaded into a MySQL database using two perl scripts. The database basically consisted of three tables, alert, scan_sum and scan_detail.

Alert stored information on each alert, in separate fields. Each field was indexed to accelerate searching. The aler t table consisted of seven fields. sec_id was intend to be a unique identifier, based on a combination of the capture time and fraction of seconds. However due to duplicate entries in the log files this was not possible. If more time was available for th is analysis further improvements on the database would have been made. The second field was cap_date this stored the capture date in a datetime format. Then the description was stored as a varchar (255), the source and destination addresses were also sto red as strings. The source and destination ports were stored as integers.

By separating each alert into its various components extracting information from the logs was then straightforward using SQL statements. This was usually extracted into text files which could then be opened and manipulated as needed in star office calc.

The scan_sum table was used to store the summary information about each scan. This included the scanning host, date and time, scan duration, number of hosts scanned and total of UDP and TCP packets. The scan_detail table was used to store the details about each scan packet. This included the capture date, source and destination address, source and destination ports and the protocol. This could then be correlated back to the scan summaries based upon the capture date and source address.

The scripts used top load the data separated the spp_portscan information and loaded the scan complete information into the scan_sum table. The rest of the alerts were loaded into the alert table. A listing of the scripts used is shown below.

load_scans.pl, takes a single scan file as a parameter.

```
#!/usr/bin/perl

use DBI();

$file_name = shift;

my $dbh = DBI->connect("DBI:mysql:database=snort;host=localhost", "mikew");
#$year = 2002;
open (ORIG, $file_name);

while ($line = readline(ORIG)) {
        if ($line =~ /\*{5,}/) {
        } elsif ($line =~ /Snort.*Report/) {
                $line =~ /\s(20\d\d)/;
                $year = $1;
        } else {
                #print $line;
                $line =~ /^(\w{3})\s(\d\d)\s(\d\d):(\d\d):(\d\d)/;
                $mt = $1;
                $dd = $2;
```

```
                $hh = $3;
                $mm = $4;
                $ss = $5;
                if ($mt =~ /[j|J]an/) {$mth = 1;
                } elsif ($mt =~ /[F|f]eb/) {$mth = 2;
                } elsif ($mt =~ /[M|m]ar/) {$mth = 3;
                } elsif ($mt =~ /[A|a]pr/) {$mth =4;
                } elsif ($mt =~ /[M|m]ay/) {$mth = 5;
                } elsif ($mt =~ /[J|j]un/)  {$mth = 6;
                } elsif ($mt =~ /[J|j]ul/) {$mth =7;
                } elsif ($mt =~ /[A|a]ug/) {$mth = 8;
                } elsif ($mt =~ /[S|s]ep/) {$mth = 9;
                } elsif ($mt =~ /[O|o]ct/) {$mth =10;
                } elsif ($mt =~ /[N|n]ov/) {$mth = 11;
                } elsif ($mt =~ /[D|d]ec/) {$mth = 12;
                }

                $line =~ /\s(.{1,3}\..{1,3}\..{1,3}\..{1,3})\:(\d{1,5})\s-
>\s(.{1,3}\..{1,3}\..{1,3}\..{1,3})\:(\d{1,5})\s(.*)/;
                $src = $1;
                $sport = $2;
                $dst = $3;
                $dport = $4;
                $proto = $5;
                my $query = sprintf ("INSERT INTO scan_detail VALUES   \(\'%04d\-
%02d\-%02d %02d\:%02d\:%02d\', \'%s\', %d,\'%s\', %d, \'%s\'\)",$year, $mth, $dd,
$hh, $mm, $ss, $src, $sport, $dst, $dport, $proto);
                $dbh->do($query);
                #print $query;
        }
}
$dbh->disconnect();
close (ORIG);
```

load_alert.pl, takes a single alert file as a pa rameter and loads it into the snort database.

```
#!/usr/bin/perl

use DBI();

$file_name = shift;

my $dbh = DBI->connect("DBI:mysql:database=snort;host=localhost", "mikew");
#$year = 2002;
open (ORIG, $file_name);

$scount = 0;
$acount = 0;
while ($line = re adline(ORIG)) {
```

```
        #clean the header information
        if ($line =~ /\*{5,}/) {
        # get the year the log was made  note this is not y3k compliant
        # and  will not work for logs made in 19 --
        } elsif ($line =~ /Snort/) {
                $line =~ /\s(20\d\d)/;
                $year = $1;
        #check for end of portscan details
        } elsif ($line =~ /End of portscan/){
                #print $line;
                $line =~ /^(\d\d)\/(\d\d)\-(\d\d):(\d\d):(\d\d)\.(\d{1,6})\s/;
                $mth = $1;
                $dd = $2;
                $hh = $3;
                $mm = $4;
                $ss = $5;
                $ss_dec = $6;
                $line =~ /\]\s(.{3,})\s\[/;
                $desc = $1;
                $line
=~/.*\s(.{1,3}\..{1,3}\.\d{1,3}\.\d{1,3}).*\stime\((\d{1,}).*\shosts\((\d{1,}).*\sTCP\((
\d{1,}).*\sUDP\((\d{1,}).*/;
                $src = $1;
                $time = $2;
                $hosts = $3;
                $tcp = $4;
                $udp = $5;
                my $query = sprintf ("INSERT INTO scan_sum VA  LUES
(\'%04d%02d%02d%02d%02d%02d.%d \', \'%04d\-%02d\-%02d
%02d\:%02d\:%02d\',\'%s\', \'%s\', %d, %d, %d, %d)",$year, $mth, $dd, $hh,
$mm,$ss, $ss_dec, $year, $mth, $dd, $hh, $mm, $ss, $desc, $src, $time, $hosts, $tcp,
$udp);
                $dbh->do($query);
                $scount= $scount + 1;
                #print $line;
                #print $query, "\n\n";
        # skip the line if it is just scan status info
        } elsif ($line=~ /spp_portscan/){
        # check if a ipaddress with a port is given
        } elsif ($line=~ /.{1,3} \..{1,3}\..{1,3}\..{1,3}\:\d{1,5}/){
                $line =~ /^(\d\d)\/(\d\d)\-(\d\d):(\d\d):(\d\d)\.(\d{1,6})\s/;
                $mth = $1;
                $dd = $2;
                $hh = $3;
                $mm = $4;
                $ss = $5;
                $ss_dec = $6;
                $line =~ /\]\s(.{3,})\s\[/;
                $desc = $1;
                $line =~ /.*\s(.{1,3}\..{1,3}\..{1,3}\..{1,3})\:(\d{1,5})\s-
```

```
>\s(.{1,3}\..{1,3}\..{1,3}\..{1,3})\:(\d{1,5})/;
                $src = $1;
                $sport = $2;
                $dst = $3;
                $dport = $4;
                my $query = sprintf ("INSERT INTO alert VALUES
(\'%04d%02d%02d%02d%02d%02d.%d \', \'%04d\-%02d\-%02d
%02d\:%02d\:%02d\', \'%s\', \'%s\', %d, \'%s\',%d)",$year, $mth, $dd,  $hh, $mm, $ss,
$ss_dec,$year, $mth, $dd, $hh, $mm, $ss, $desc, $src, $sport, $dst, $dport);
                $dbh->do($query);
                $acount= $acount + 1;
                #print $line;
                #print $query, "\n\n";
        #if nothing else just load the src & destination address and the description
        } else {
                $line =~ /^(\d\d)\/(\d\d)\-(\d\d):(\d\d):(\d\d)\.(\d{1,6})\s/;
                $mth = $1;
                $dd = $2;
                $hh = $3;
                $mm = $4;
                $ss = $5;
                $ss_dec = $6;
                $line =~ /\]\s(.{3,})\s\[/;
                $desc = $1;
                $line =~ /.*\s(.{1,3}\..{1,3}\..{1,3}\..{1,3})\s-
>\s(.{1,3}\..{1,3}\..{1,3}\..{1,3})/;
                $src = $1;
                $dst = $2;
                my $query = sprintf ("INSERT INTO alert VALUES
(\'%04d%02d%02d%02d%02d%02d.%d \', \'%04d\-%02d\-%02d
%02d\:%02d\:%02d\', \'%s\', \'%s\',0,\'%s\',0)",$year, $mth, $dd, $hh, $mm, $ss,
$ss_dec, $year, $mth, $dd, $hh, $mm, $ss, $desc, $src, $dst);
                $dbh->do($query);
                $acount= $acount + 1;
                #print $line;
                #print $query, "\n\n";
        }
}
print $acount, "  Records added to alert \n", $scount, "  Records added to scan_sum \n";
$dbh->disconnect();
close (ORIG);
```

References

RFP (2001) *libwhisker function reference*. http://www.wiretrip.net/rfp/p/doc.asp/i2/d72.htm#fn -anti_ids#

CERT (2002). *Vulnerability Note VU#107186 Multiple vulnerabilities in SNMPv1 trap handling*. http://www.kb.cert.org/vuls/id/107186

Northcutt, S. Cooper, M. Fearnow, M. Frederick, K. (2001). *Intrusion Signatures and Analysis*. USA, New Riders.

SANS (2002) *Network Traffic Analysis using tcpdump parts 1 and 2*. USA SANS Institute

SANS (2002) *TCP/IP for Intrusion Detection*. USA SANS Institute

SANS (2002) *IDS Signatures and analysis parts 1 and 2*. USA SANS Institute

Comer, D. (2000). *Internetworking with TCP/IP Principles protocols and architectures* . USA. Prentice Hall

Roesch, M. (2001) *Snort Users Manual Snort Release: 1.8.2*. [on-line] http://www.snort.org

Various (2002) Snortrules.tar.gz [on -line] http://www.snort.org