10

THE THREAT FROM WITHIN – AN ANALYSIS OF ATTACKS ON AN INTERNAL NETWORK

John Haggerty, Qi Shi, and Madjid Merabti
Liverpool John Moores University, School of Computing & Mathematical Sciences, Byrom Street, Liverpool, L3 3AF. E-mail: cmsjhagg@livjm.ac.uk, q.shi@livjm.ac.uk, m.merabti@livjm.ac.uk

Abstract: Computer security, whether protecting from external or internal attacks, is a major concern with our reliance on networks for the flow of information between users and organisations. This paper summarises research into the threat to an internal network. The case study, which includes a denial of service attack and a worm infection, allows us to address the effectiveness of our security countermeasures and how they are affected when an incident takes place in the wild. Through the lessons learned, we can address our security to provide better defence mechanisms for the future. This research also provides us with new research opportunities, such as the problems of key security components as attack targets or the use of traffic monitoring to provide defences against particular network-based attacks.

1. INTRODUCTION

Computer security is a major concern in today's interconnected world with business' reliance on information flow for competitive advantage. Whilst networking technology facilitates user access to a large number of information resources, it also opens up the organisation to a wide number of threats [5]. Organisations realise that they are facing a dichotomy. On the one hand, they must be interconnected to take advantage of information flow; on the other, they must secure their systems in the face of the number of threats and problems that networking harbours. These threats are not only from outside networks. It is recognised that there is a substantial threat from within the organisation, as demonstrated by the Computer Security Institute/Federal Bureau of Investigation annual report [10].

This paper summarises research into the threat to an internal network. The aim of the research was to monitor traffic to assess seasonal changes in security incidents due to new users on the network. During the research period, a number of security incidents came to light and were captured for

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: 10.1007/978-0-387-35586-3_46

M. A. Ghonaimy et al. (eds.), *Security in the Information Society*

analysis. These included a major denial of service incident and a worm infection on the network, which form the basis of this paper. Analysis of these incidents provides us with further insight into how we are better able to secure our systems for the future. These key findings are presented within this paper.

The research also tested Mansfield *et al*'s assertion that one major problem in defending against denial of service attacks is to distinguish between normal and abnormal traffic [6]. Therefore, the understanding of what can be considered normal traffic on the network had to be found.

Within this paper, we discuss a case study of network security incidents on an internal network. This analysis covers two important issues in security research that has not been addressed satisfactorily in existing literature. First, we are able to see the real life effectiveness of security countermeasures and how they are affected when an incident takes place in the wild. Second, this real case provides us with opportunities to examine the effectiveness of existing technologies and propose research topics, which require urgent effort to develop more cost-effective solutions to these real problems. The experiment details within this paper allow us to see threats and attacks to our networks. From our analysis of these details, we are able to learn a number of lessons that help us secure our systems against further attacks. From these lessons, we are able to present new areas of research, such as the use of network analysis for key attack signatures.

This paper is organised as follows. Section 2 presents the methodology of the research and the key decisions made. Section 3 presents the key findings and issues related to positive and false positive attackers, and the attacks themselves. Network signatures of the attacks are presented. Section 4 provides an analysis of the experiment findings and a network denial of service incident observed. Section 5 presents lessons learned from the experiment and proposed future research for improved network security. Finally, we make our conclusions.

2. METHODOLOGY

A four-week period was chosen for the research, which provided a reliable baseline for determining normal from abnormal traffic. The operating system (OS), Windows 98 SE, was chosen as the network has many Windows based machines used by clients. The network had recently moved over to Windows 2000, but a number of previous Microsoft OS remained in service by users. There were two key benefits of this choice; a Windows 9*x* client would stand out in a network scan using OS fingerprinting, and there are a number of known security vulnerabilities of this OS compared to the latter. It was thought that if a network scan was conducted by a malicious person, the machine would be a more attractive target.

BlackICE Defender was chosen as the Intrusion Detection System (IDS) to be used on the control machine for the duration of the research for two reasons. First, an IDS had to be chosen that ran on Microsoft OSs. BlackICE Defender is commercially available and provides attack warnings and information about network traffic levels. Second, the machine had to look like a users' machine. Whilst a UNIX based IDS such as Snort is widely available and used, if a malicious person was able to gain access to the computer, the presence of such an IDS may alert them to the machine's intent. Therefore, software that a home user may use was more appropriate.

The IDS records attacks based on its signature database and writes any noted attacks to an evidence log. This log provides details of the packet contents. However, one down side of the software is that there is no means of reading the log contents of the .enc files that are created. Therefore, a network traffic analyser had to be used. A network analysis tool, Analyzer, was used that was able to read the .enc attack.[1] The logs once opened in this program allowed analysis of suspicious traffic.

BlackICE allows the user to delete attackers from the attacker records by deeming them trusted. However, for the duration of the research, all reported attackers that generated suspicious traffic were recorded. Once an attacker was recorded by the system, they were investigated by identifying and verifying the host and by analysing the traffic in the attack log. This allowed the attacks to be recorded as either positive or false positive attack detections.

3. EXPERIMENT FINDINGS

Over the four-week research period, a total of 1493 attacks against the control computer were reported by the IDS. These attacks are broken down as follows:

- 1084 false positives
- 409 positive attacks
- 8 types of attack
- 20 attackers

Within this section, we look at these key findings in more detail. First, we explore the positive and false positive attacks reported by the IDS. Second, we present the variety of attacks that were reported. Third, we present the issue of determining a 'real' attacker from those reported attackers. Finally, we present the attack signatures noted in network traffic of a real incident, that of a Code Red worm infection.

3.1 Positive and False Positive Attacks

The root of all intrusion detection is based in analysing a set of discrete, time-sequenced events for patterns of misuse [11]. IDSs, whether

[1] This tool is available at http://netgroup-serv.polito.it

host-based or network-based, identify Events of Interest (EOI) within the system [8]. These are possible attacks against the target machine or network. There are three main issues surrounding EOIs; balance between false positives and false negatives, ensuring that EOIs are detected, and the limits on the system's ability to detect attacks [8]. Within our research we identified a number of positives and false positives. A *positive* is when the recorded attack equates to an actual EOI. A *false positive* is when an event is recorded as an attack but is not.

A number of events were recorded by the IDS that on further analysis were found not to be deliberate attacks. For example, a network print server routinely sent out Simple Network Management Protocol (SNMP) *discovery broadcast* traffic to the network to ascertain clients that were available. Also, a file server routinely sent out SNMP traffic, and these were recorded as User Datagram Protocol (UDP) port probes on the target system.

The network activity reported as attacks by the IDS are illustrated in figure 1 below. False positives are those recorded attacks that on later analysis proved to be normal network activity. The positives are all those recorded attacks that proved to be malicious events. As can be seen in the chart, positive attacks occurred only in weeks two and four. These were due to two Code Red worm infections in those weeks, which will be discussed in section 4. The Code Red worm is a self-propagating malicious code that exploits vulnerabilities in Microsoft Web servers [3].

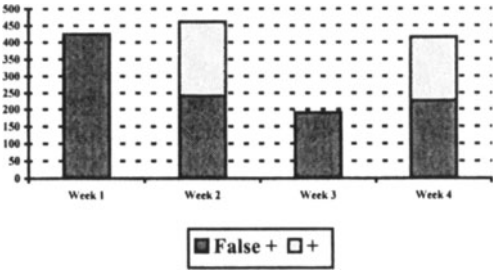

Figure 1. *Total number of incidents reported during the research period. False positive and positive attacks are broken down.*

3.2 Types of Reported Attack

Throughout the research period, a variety of attacks were reported. These were: UDP probe, SNMP probe, SNMP discovery broadcast, Transmission Control Protocol (TCP) scan, TCP probe, Hyper-Text Transfer Protocol (HTTP) probe, Telnet probe, and a suspicious duplicate address. Of these reported attacks, UDP probe, SNMP probe, SNMP discovery broadcast, HTTP probe and TCP scan were the most popular, as can be seen in figure 2 below.

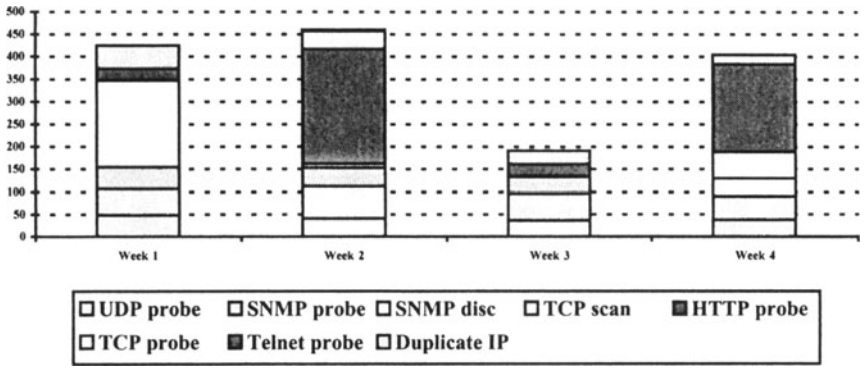

Figure 2. *Types of attack reported. The increase in HTTP port probes caused by the Code Red worm can be clearly seen during weeks two and four.*

Despite the number of attacks reported by the IDS, most remain false positives. During the two periods of network Code Red infection a large amount of HTTP port scans were detected, providing a signature for the virus' presence within the network.

3.3 Attackers

As can be seen by the total number of attacks reported by the IDS versus the number of false positives, many of the security incidents flagged by the control system were not security incidents at all. The need for accuracy of the IDS is crucial to provide protection to a system [12]. However, the research has identified the need for both the system and those analysing the results to have an understanding of the system the IDS is protecting. This enables the analyst to quickly identify positive attacks from false positives, and therefore react to an actual event.

A number of servers on the network sent out packets that were reported as attacks but were not. For example, the server hpov sent out a large number of SNMP port probes but was identified as a network management server. Reported attempts of intrusion by this machine included scans of UDP port 161 (SNMP), TCP port 280 (HTTP management), and TCP port 80 (HTTP client request). The server CMS3 was recorded as making intrusion attempts when it sent UDP packets to port 161 on the broadcast address 255.255.255.255. CMS6, another network server, also probed the network for SNMP enabled machines. The IDS flagged SNMP traffic as possible intrusions as it is designed for connections with the Internet rather than internal networks behind a firewall. Due to the information that SNMP provides to an outside attacker about network configurations, port 161 TCP and UDP should be blocked from the Internet [8]. However, on an internal network, these services are required for network management. CMS6 also attempted connections to the control machine via TCP port 80, port 23 (Telnet), and port 2039 (normally associated with Sun Microsystems Network File System operations). On further investigations

into these machines, this and other traffic were deemed to be normal network operations.

Figure 3 below shows the reported attacks by attacker over the research period. A number of servers were reported throughout the period as being attackers. However, during weeks two and four a greater number of attackers were seen, due to the Code Red infection within the system. These infected machines are indicated by asterisk in the legend. This is correlated with the large number of HTTP probes seen in figure 2, a signature of the virus' presence on the network.

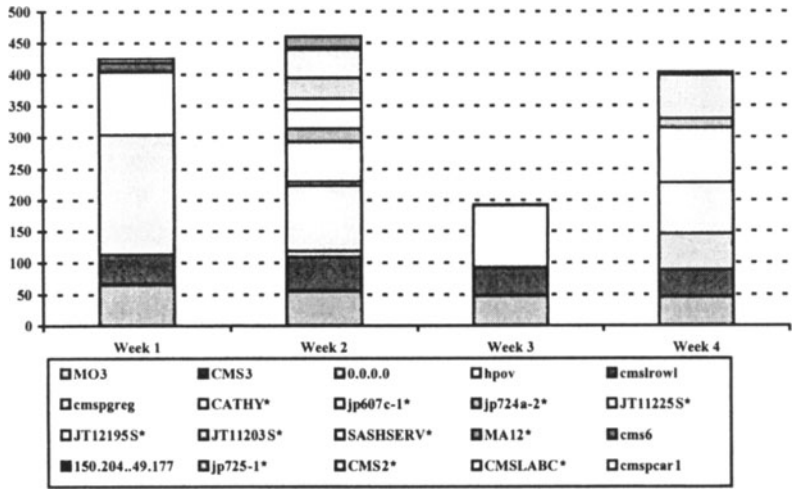

Figure 3. *Attackers reported during the research period.*

3.4 Code Red Network Signatures

The Code Red worm attempts to connect to TCP port 80 on a randomly chosen host assuming that a web server will be found. Upon a successful connection to port 80, the attacking host sends a crafted HTTP GET request to the victim, attempting to exploit a buffer overflow. If the exploit is successful, the worm begins executing on the victim host [3]. During the research period, two infections by the worm were observed. The second infection caused massive network disruption despite being less severe than the first.

Traditionally in the detection of malicious programs, software is able to detect the malicious program on the infected machine by identifying patterns matching entries within its database of known signatures. However, during the attacks seen during the research period, a number of signatures were identified within the network traffic, which allowed us to identify the worm. This enabled the quick identification of infected machines and to halt

the spread of the worm further in the network. These signatures were as follows:

Network traffic increase and HTTP port requests. Once the worm has infected a machine, it seeks to infect other machines on a trigger time. It does this by exploiting the vulnerability in other machines that it used with the infected machine. Its exploitation of the vulnerability relies on other machines on the network running similar software to its own; IIS Web servers. The network that was infected sat behind a firewall, and there should be very few internal Web servers. However, a number of the machines had not been configured properly, so a number of services that were not required were running, including Web servers, allowing the spread of the worm. As the infected machines attempted to infect others, an upsurge in the number of HTTP port requests generated network traffic at the times when the worm was triggered. This traffic can be seen in figure 4 below.

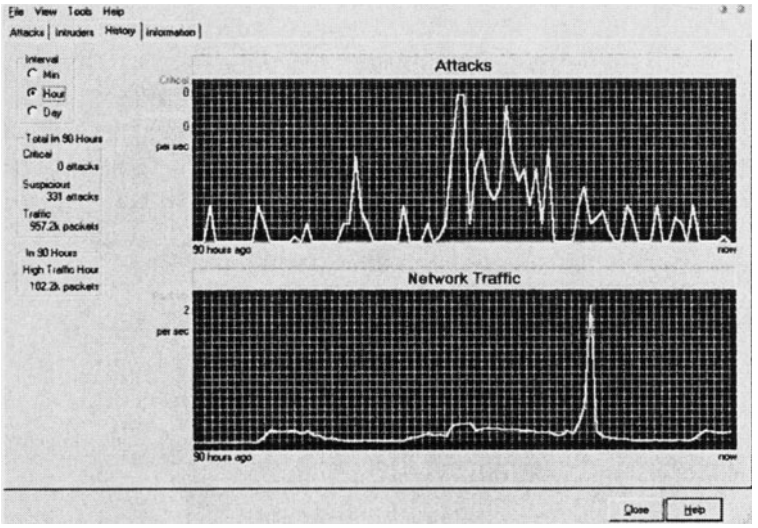

Figure 4. *Upsurge in network traffic recorded by the IDS during week four. The top line shows the number of attacks against the target system. The bottom line shows the upsurge in network traffic caused by the number of HTTP port probes.*

The IDS creates logs for audit purposes. On analysis of these logs, we can also see the traffic generated by infected machines. As we can see in figure 5, the audit logs show the large number of HTTP port requests.

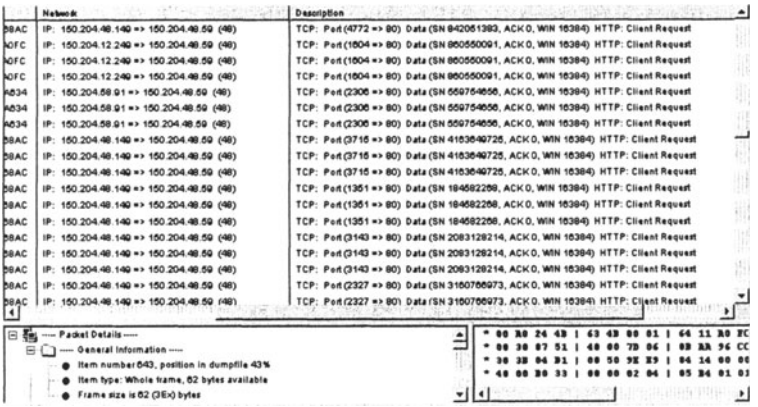

Figure 5. *IDS log showing the number of HTTP port requests generated by the worm visible in the top right column.*

Uniform packet size. Analysis of the individual packets generated by the worm show that they were of a uniform size. These packets, as illustrated in figure 6, were HTTP client requests of 62 bytes. In comparison, the HTTP client requests made by other machines on the network were 60 bytes, as illustrated in figure 7. Therefore, the combination of the number of packets and their uniform size indicates Code Red infection.

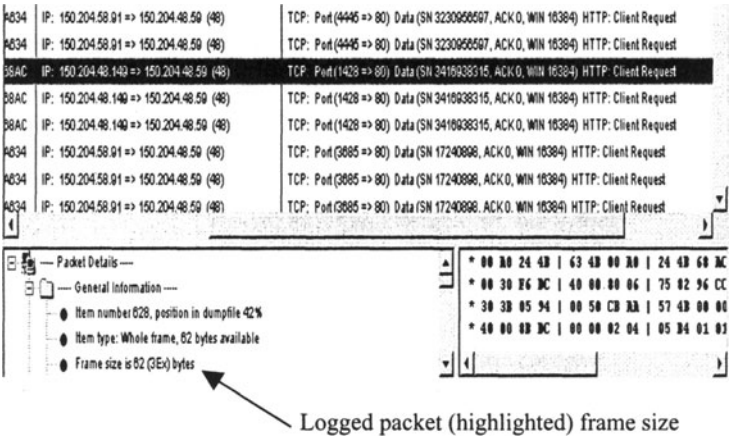

Figure 6. *Payload of a Code Red attack packet from one of the infected machines. The top of the diagram shows the packet within the log. The bottom half, as indicated, shows the individual packet details.*

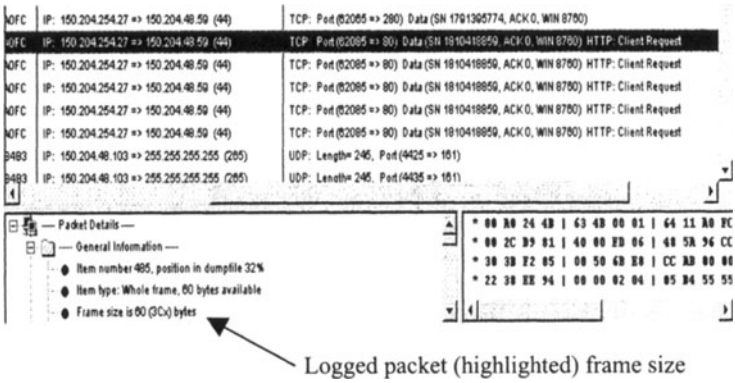

Figure 7. *Payload of a HTTP client request from a machine not infected with Code Red. The top of the diagram shows the packet within the log. The bottom half, as indicated, shows the individual packet details.*

Randomised ports. The attack packets in the audit log consisted of 3 packets from one port number to port 80 on the intended target. Once 3 packets were sent, another random port number was used by the infected machine to send the next HTTP request packet. This pattern was observed on all infected machines during the attacks.

Therefore, as we can see, the traffic caused by the presence of the virus within the network has a number of tell-tale signatures, that once combined, indicates the presence of the problem on the network. This enables us to reduce the number of false positives recorded by the IDS and quickly isolate those machines that will need to be patched from uninfected machines. This would be faster than attempting to patch all machines on the network, infected or not, through virus scanner updates or downloaded patches.

4. ANALYSIS OF THE EXPERIMENT

Denial of service attacks are when a legal network user is prevented from performing his/her functions or prevents other users from performing their functions [7]. Denial of service attacks often overwhelm the victim host or network to the point of unresponsiveness to the legitimate user [11]. With the reliance of organisations on networks and interconnectivity, these attacks can have a serious effect. These effects are at all levels of network size. For example, an attack on a single host, such as a home user, may prevent a transaction from taking place. At the Local Area Network (LAN) level, an organisation may be prevented from conducting its business with key elements of the LAN infrastructure.

There were two incidents that were observed during the research that pertained to denial of service. The first was caused by ICMP traffic generated in reply to messages from misconfigured servers. The second was caused by the Code Red worm within the system.

Throughout the research, a large amount of traffic was recorded by the IDS as originating from two servers within the network; MO3 and CMS3. These servers provide application and print services to the network users. As we have seen during the research, in their normal operations, these servers send out SNMP traffic via UDP to port 161 to discover connected network nodes. This traffic was recorded by the IDS as UDP port probes. The target machine, not running SNMP services, received these requests and then replied with ICMP *destination unreachable* messages. IP is not designed to be absolutely reliable and it therefore assumes that the network will experience problems in routing data from source to destination. The purpose of ICMP messages is to provide feedback about problems in the communication environment, not to make IP reliable [9]. Whilst ICMP performs essential network functions, they can also cause problems in communications in certain circumstances [1, 2]. A large amount of needless traffic was generated on the network by MO3 and CMS3 sending out requests and then receiving these ICMP return messages from other similar machines. This caused some degradation of service in the two servers.

The second incident was caused by the infection of a number of machines with the Code Red worm. The first infection occurred in week 2 and a number of machines were noted as sending out a large amount of HTTP client requests to machines within the network. The 8 infected machines were identified and patches were then applied by the system administrators to remove the infection in accordance with security alerts about the problem. In total, 219 of these HTTP requests were received by the control machine before patches could be applied. With approximately 330 machines within the network, we can see that a large amount of network traffic was generated with requests made to each and every machine. However, services were not seen to greatly degraded for two reasons; a recent upgrade in the network bandwidth and the students having not fully returned.

During week 4, the Code Red worm in an infected machine was triggered. This machine had been infected during week 2 and patched unsuccessfully. When the worm was triggered, it began sending messages to both other machines on the network to spread the infection and also tried to connect to an external IP address. In total, 190 HTTP connection requests were observed against the control machine originating from 4 machines. Only one of the machines had been part of the first Code Red infection. These machines were quickly identified and taken offline until patches could again be applied.

This second attack resulted in severe network disruption. Whilst the attack traffic levels observed were less than the attack in week 2, the entire university network ground to a halt. As the original infected machine attempted to connect to the Internet, it began sending traffic to the firewall to forward on to the external IP address. The firewall logged all attempts as

suspicious and did not forward the traffic. The number of requests caused the firewall's hard disks to fill with audit information, until the point that the entire disk became full and the firewall crashed ensuring that no traffic could pass through it. When the firewall was finally brought back on line, three routers crashed, which caused further disruptions during the day within the network. However, in the early afternoon, attack traffic was again launched by the worm and the infected domain had to be taken offline until the problem was rectified.

The denial of service was not necessarily caused by the Code Red worm traffic within the networks, although the amount of traffic would have seriously degraded network services. This attack involved less infected machines and a lower amount of attack traffic than the first incident in week 2. The serious network disruptions were caused by a misconfigured main firewall that filled its hard disks with audit information. This may have been caused for two reasons; a large amount of traffic in a short period of time was recorded by the main firewall, or the audit logs are not rotated or deleted within a suitable time period. In an earlier firewall implementation by the authors on a separate network domain, it was noted that in a short period of time, audit logs in firewalls will fill quickly if they are not managed properly. In this firewall, the audit files were reviewed so that the audit data was compressed on a five-day rotation and old files deleted regularly if not needed. This ensured that the hard disks would not fill. In the main firewall, this appears to have not been done, so data would eventually grow to fill the hard disks, causing the machine to become inoperable. The vulnerability of key security elements to this type of issue has been highlighted in IDS, which have been identified as key targets [12]. For example, the "Stick" distributed denial of service tool specifically targets IDSs by sending spurious packets to fill up event logs [4]. However, this research has highlighted that this also remains a vulnerability in other security devices such as firewalls.

5. LESSONS LEARNED AND FUTURE RESEARCH

Below is a summary of the key lessons learned from the research:

Patches. Ensure patches are applied properly. In both incidents, one machine was used by the worm to infect others on the network.

Attack escalation. One type of attack can escalate into another. At first, the worm infection remained just that. However, a culmination of factors escalated the situation into major network disruptions, bringing down the entire network.

Policies and Planning. There is a need for sound policies and planning. When the firewall crashed, there was no back-up procedure in place to allow network operations to continue as normal. Thus, network services were denied until the problem was rectified. Policies and plans must be put in place to prepare for such a problem.

Key targets. Security components are key targets. By sending spurious packets to deliberately fill audit logs on key security components, a serious security compromise can be achieved by a malicious attacker.

Firewall configuration. The firewall is an essential security network element and must be configured correctly to ensure its operation and provide the protection it offers. With audit logs filling up the hard drive, the firewall became inoperable. This vulnerability must be guarded against.

Insider threat. The threat from within is not always from malicious attackers. The research attempted to test the hypothesis that network security incidents were likely to increase as more users used the system with the students returning. However, the attacks observed were unable to substantiate this assertion.

Signatures. The signature of the worm can be ascertained by network monitoring. Normally, a virus checker on user desktops would detect an infection. However, these are retrospective and only as good as the last update. Analysis of the network traffic allowed timely identification of worm infected machines, and the network traffic showed the type of infection.

From these lessons we can see that two issues require further research. First, the issue of security components being the target of successful attacks. Whilst this issue has been addressed in areas such IDSs, we can see from our denial of service incident that this is a very real problem. Second, network monitoring for attack detection and containment. Many security components use signature matching of traffic against particular ports to detect and react to attacks. However, this only a part solution to problems such as denial of service where network traffic monitoring will play a larger part in providing a defence to an attack.

6. CONCLUSIONS AND FURTHER WORK

Computer security, whether protecting from external or internal attacks, is a major concern with our reliance on networks for the flow of information between users and organisations. During this research into a university network, we were unable to affirm the hypothesis that security incidents were likely to increase as more users returned from their summer break. However, we were able to capture and analyse a number of other security concerns, including a worm infection of the network and a severe denial of service incident.

A number of lessons have been learned from the research. First, that key security components such as firewalls are susceptible to attack that remove the defence measures that they provide. This is achieved by targeting their mechanisms, such as audit logs, to fill disk space and cause them to crash, as was seen in our denial of service incident. The problem is compounded if these key security components are not configured with this problem in mind. Second, we are able to quickly identify major security incidents, such as worms, in a timely manner through an understanding and

monitoring of the traffic over the network. Finally, the threat from within the network is not always from malicious attackers. Misconfigured computers are likely to cause security problems unless they are identified and reconfigured.

The case study presented in this paper allows us to address issues that have not been satisfactorily covered in existing literature. We are able to see the real life effectiveness of security countermeasures and how they are affected when an incident takes place in the wild. Second, this real case provides us with opportunities to examine the effectiveness of existing technologies and propose new research topics, such as the problems of key security components as attack targets or the use of traffic monitoring to provide defences against particular network-based attacks. Our further work into denial of service intends to draw on this experience to provide countermeasures to a number of attacks.

REFERENCES

[1] Baltatu, M., Lioy, A., Maino, F. & Mazzocchi, D., "Security Issues in Control, Management and Routing Protocols," *Computer Networks*, vol. 34, pp. 881-894, 2000.

[2] Bellovin, S. M., "Security Problems in the TCP/IP Protocol Suite," *Computer Communications Review*, vol. 19, pp. 32-48, 1989.

[3] CERT, "CERT Advisory CA-2001-19 'Code Red Worm Exploiting Buffer Overflow in IIS Indexing Service DLL," CERT Advisory, http://www.cert.org/advisories/CA-2001-19.html, download 2001, 2001.

[4] Howard, S., "Stick and Network Signature Based Intrusion Detection", SANS Institute Info Sy Reading Room Technical Report, http://www.sans.org/infosecFAQ/threats/stick.htm, download 2001, 11 April 2001.

[5] Lin, A. & Brown, R., "The application of security policy to role-based access control and the common data security architecture," *Computer Communications*, vol. 23, pp. 1584-1593, 2000.

[6] Mansfield, G., Ohta, K., Takei, Y., Kato, N., & Nemoto, Y., "Towards trapping wily intruders in the large," *Computer Networks*, vol. 34, pp. 659-670, 2000.

[7] Muftic, S., Patel, A., Sanders, P., Colon, R., Heijnsdijk, J. & Pulkkinen, U., *Security Architecture in Open Distributed Systems,* John Wiley & Sons, Bath, UK, 1993.

[8] Northcutt, S., *Network Intrusion An Analyst's Handbook,* New Rider Publishing, USA, 1999.

[9] Postel, J., "RFC792 Internet Control Message Protocol," The Internet Society Technical Report, http://www.cis.ohio-state.edu/htbin/rfc/rfc792.html, download 2000, 1981.

[10] Power, R., "2001 CSI/FBI Computer Crime and Security Survey", Computer Security Institute/Federal Bureau of Investigation Technical Report, vol. 7, no. 1, Spring 2001.

[11] Proctor, P. E., *The Practical Intrusion Detection Handbook*, Prentice Hall, Upper Saddle River, NJ, 2001.

[12] Ptacek, T. H. & Newsham, T. N., "Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection," Secure Networks Inc. Technical Report, http://www.clark.net/~roesch/idspaper.html, download 2001, January, 1998.