Global Information Assurance Certification Paper

Mydoom in the Case of the Unsuspecting Vendor

GIAC Certified Incident Handler (GCIH)

Practical Assignment Version 3

Chris Chromiak

SANS Mardi Gras February 2004

Submitted: June 14, 2004

ACKNOWLEDGEMENTS

Mom and Dad; without your struggles and sacrifices throughout your entire lives this would not have been possible. Words will never be able to express how grateful I am to both of you. I love you both.

Big Sis; you're not so bad either. Thanks a million for coming back home when Dad became ill. It meant the world to all of us and your help will always be remembered.

Chris Hyndman; I really miss you buddy…..we all do.

Krzysio

ABSTRACT

This paper will explain how the virus W32.Mydoom.A@mm (referred to as Mydoom throughout most of the paper) found its way into a fictitious corporate network and was able to spread as rapidly as it did. In this particular scenario, Mydoom is initiated from an unsuspecting vendor who is also a competitor in the ABC market space with the company 'Viruses Unlimited'. Although corporate policies exist they are often not enforced, as was the case this time. Salesperson, 'Joe Infect' (who looked very professional) was granted access into the 'Viruses Unlimited' office without any hesitation from the receptionist after he told her that he had a meeting with the CEO, 'John Doubleclick'. The receptionist was even so kind as to escort Joe to the executive meeting room and show him where the data ports were so that he could plug into the corporate network and check his email. Little did the Incident Response team know when they began their day that they would be in full Incident Handling mode before the day was done. Each phase of the Incident Handling process was used by those brave 'firefighters' we know as Incident Handlers in the long hours that were spent on cleaning up the Mydoom infection.

Note: Although this infection is based on a fictitious corporate network, many of the phases in the 'Stages of Attack' and 'Incident Handling' sections are based on personal experiences of my own. You may be surprised at how easy it is to launch an attack inside of a network!

TABLE OF CONTENTS

1.0 STATEMENT OF PURPOSE 5

2.0 THE EXPLOIT 6

2.1. NAME 6
2.2. OPERATING SYSTEMS 6
2.3. PROTOCOLS/SERVICES/APPLICATIONS 7
2.3.1. Email 7
2.3.2. Kazaa 7
2.3.3. Transmission Control Protocol (TCP) 8
2.3.4. Domain Name System (DNS) 8
2.3.5. Hypertext Transfer Protocol (HTTP) 9
2.4. VARIANTS 9
2.5. DESCRIPTION 11
2.6. SIGNATURES OF THE ATTACK 12

3.0 THE PLATFORMS/ENVIRONMENTS 17

3.1. VICTIM'S PLATFORM 18
3.2. ATTACKING SYSTEM 18
3.3. TARGET SYSTEMS 19

4.0 STAGES OF THE ATTACK 19

4.1. RECONNAISSANCE 19
4.2. SCANNING 21
4.3. EXPLOITING THE SYSTEM 22
4.4. KEEPING ACCESS 29
4.5. COVERING TRACKS 29

5.0 THE INCIDENT HANDLING PROCESS 30

5.1. PREPARATION 30
5.2. IDENTIFICATION 44
5.3. CONTAINMENT 49
5.4. ERADICATION 51
5.5. RECOVERY 55
5.6. LESSONS LEARNED 58

6.0 REFERENCES 60

APPENDIX A 63

APPENDIX B 65

APPENDIX C 67

1.0 STATEMENT OF PURPOSE

It did not take long into the year of 2004 before the IT Security community had its 'hair on fire' with the rapid spread of Mydoom. On January 26, 2004, a virus known as Mydoom spread across the Internet at record speed. All it took was the typical uneducated home or corporate user to double-click on an attachment received via email or downloaded while file sharing with Kazaa.

This paper will focus on the spread of Mydoom after an infected email attachment is opened, the various platforms and environments affected, the stages of the attack and the incident handling process on the Viruses Unlimited corporate network. This particular outbreak was possible because of weak security policy enforcement and lack of user awareness. It was not the social engineering of Mydoom that caused chaos among the network, but the social engineering of the attacker who was able to 'weasel' his way into the corporate office and spread the virus.

Due to the network outage and overtime hours spent on cleaning up the Mydoom infection the Viruses Unlimited corporation was dealt a tough hand in the 'lessons learned' department. After this infection, Senior Executives realized just how important user education, policy enforcement and physical security are to the organization. Even though anti-virus vendors did not have a signature file ready when the outbreak hit, Viruses Unlimited painfully learned that if their corporate security policies were enforced then the spread of Mydoom could have been avoided. Signature file updates would have been pushed down from the corporate anti-virus server when they became available from the vendor.

2.0 THE EXPLOIT

2.1. Name

The name of the exploit in question is W32.Mydoom.A@mm [1], which will also be referred to as Mydoom throughout the paper. Mydoom will only spread if two conditions are met; a user opens a malicious attachment and that same user has out of date anti-virus definition files (older than January 26, 2004 in this case).

Different anti-virus vendors use their own proprietary names for Mydoom even though they all refer to the same virus (Table 1). Therefore, any one virus can be referred to by multiple names.

Vendor	Name
Symantec	W32.Mydoom.A@mm (previously known as W32.Novarg.A@mm)
McAfee	W32/Mydoom@MM
Computer Associates	Win32.Mydoom.A
F-Secure	Mydoom (previously known as Novarg)
Sophos	W32/MyDoom-A
Trend Micro	WORM_MYDOOM.A (previously known as WORM_MIMAIL.R)
Kaspersky	I-Worm.Mydoom (previously known as I-Worm.Novarg)

Table 1 – Aliases for Mydoom

Detailed information on Mydoom can be found in the links provided to the vendor websites listed in Table 1 as well as CERT Incident Note IN-2004-01 [2].

http://www.cert.org/incident_notes/IN-2004-01.html

2.2. Operating Systems

Mydoom does not look for vulnerabilities in operating systems (OS') and applications such as missing patches or buffer overflows. It will only propagate with user interaction (i.e. double-clicking an attachment) and if the user does not have current anti-virus signatures applied. Therefore, OS version numbers and patch levels do not apply. Refer to Table 2 for OS' affected.

Operating System
Microsoft Windows 9x
Microsoft Windows ME
Microsoft Windows NT
Microsoft Windows 2000
Microsoft Windows XP
Microsoft Windows Server 2003

Table 2 – Operating Systems affected

2.3. Protocols/Services/Applications

Mydoom propagates via electronic mail (email) and Kazaa. Once a system is infected Mydoom will install its own Simple Mail Transfer Protocol (SMTP) engine, Domain Name System (DNS) client and a backdoor that listens on Transmission Control Protocol (TCP) ports 3127-3198. In addition, infected systems may try a DOS against www.sco.com beginning February 1, 2004 and ending February 12, 2004 [1].

2.3.1. Email

Email is generally sent using SMTP over TCP port 25 [3]. Request for Comment 2821 (RFC 2821) goes into full detail on what the purpose of SMTP is and how it works [4].

SMTP is used to send and receive email between email servers and email clients. The objective is to send mail in a reliable and timely manner. When someone sends an email, the client first sends it to the local email server, which validates the destination address and transfers the mail. If the mail server cannot validate the destination address, it will relay the mail to another mail server until it reaches its final destination. This relay from mail server to mail server will continue to happen until the recipient of the email is reached.

Mail servers that allow relaying and have no restrictions open themselves up to potential trouble. It is likely that a large amount of junk email known as spam will be sent and received by the email server. It is also likely that these messages will come from a fake or spoofed email address.

2.3.2. Kazaa

Kazaa is a file sharing service available to anyone on the Internet and uses P2P technology [5]. Peer-to-Peer networking allows all systems to essentially act as a client and a server. All Internet users who have installed the Kazaa client are able to share vast amounts of resources with each other for upload and download at no cost.

Kazaa has its own proprietary network protocol known as Fast Track [6]. It is a major threat for corporate networks and a headache for network administrators. Kazaa uses both the TCP and User Datagram Protocol (UDP) and tries port 1214 for its initial connection [7]. If port 1214 is not available it will try a port ranging from 1000-4000 and if none of those are free it uses port 80. It is very easy for a malicious user to insert malcode into a file that looks legitimate. If this file is downloaded and opened the malcode is executed and the potential for damage to a corporate network is probable.

2.3.3. Transmission Control Protocol (TCP)

TCP allows two hosts to establish a connection with each other and exchange information. It is a connection-oriented protocol, reliable and data is delivered in order to the destination. Full details on TCP can be found at RFC 793 [8]. Computers establish a connection with each other using a TCP Handshake. A TCP Handshake consists of 3 steps outlined below [8]:

a) Computer A sends a synchronization (SYN) packet
b) Computer B acknowledges (ACK) the connection attempt and sends back a SYN/ACK packet
c) Computer A acknowledges the receipt of the packet from Computer B and a connection is established

Due to the vast number of ports that TCP uses, it is simple for an attacker to pick any number of uncommon ports in their malcode and establish a connection into the internal network once exploited. When a connection is established it may be possible for an attacker to do any number of things such as send spam on your behalf.

2.3.4. Domain Name System (DNS)

The purpose of DNS is to translate domain names into Internet Protocol (IP) addresses. Domain names are much easier to remember than IP addresses because they are alphabetic as opposed to numeric. All Internet addresses are based on IP addresses, hence the need for DNS. For example, the domain name www.virusesunlimited.local translates into IP address 192.168.2.1. The RFC for DNS can be found at RFC 3658 [9]. DNS uses port 53 [3].

Mydoom installs its own DNS client on infected systems. Therefore, it does not rely on a default DNS if it is not available. It is able to use its local DNS client to resolve domain names and continue to spread itself via valid email addresses or initiate a DOS against www.sco.com.

2.3.5. Hypertext Transfer Protocol (HTTP)

HTTP is an application level protocol that communicates on port 80 and is used by the World Wide Web (WWW) [10]. Most messages and files on the WWW are transferred using HTTP. A web server waits for HTTP requests sent out by HTTP clients. Once the request comes through the web server uses a HTTP daemon to retrieve the requested web page for the client. Refer to RFC 2616 for full details on HTTP [10].

Mydoom contains a HTTP Denial of Service (DOS) attack within its code. A DOS attack causes a system to crash by overloading it with requests. Infected systems will attempt to bring down web servers at The Sco Group. If successful this will mean lost productivity, overtime hours spent on bringing other web servers online, unhappy customers and bad publicity.

2.4. Variants

Since January 26, 2004, there have been nine variants of W32.Mydoom.A@mm. These variants are outlined in Table 3. Note that the variants listed use Symantec's naming convention [11].

W32.Mydoom.A@mm characteristics: Discovered January 26, 2004

a) Attempts to perform a DOS against www.sco.com from February 1, 2004 to February 12, 2004
b) Infection length is 22 528 bytes
c) Opens a backdoor that listens on TCP ports 3127-3198
d) Propagates via mass mailing and Kazaa
e) Has its own SMTP engine

W32.Mydoom.A@mm Variants	Discovery Date	Differences from W32.Mydoom.A@mm
W32.Mydoom.B@mm	January 28, 2004	a) Attempts to perform a DOS against www.microsoft.com as well as www.sco.com that ends March 1, 2004 b) Infection length is 29 184 bytes c) Opens a backdoor on TCP ports 1080, 3128, 8080 and 10080 d) May install itself on systems infected with Mydoom.A e) Overwrites the local host file

W32.HLLW.Doomjuice	February 9, 2004	a) Only attempts to perform a DOS against www.microsoft.com b) Uses Mydoom.A infected computers to propagate c) Infection length is 36 864 bytes d) Uses TCP port 3127 to attempt to connect to randomly generated IP addresses
W32.Mydoom.dam	February 19, 2004	a) Corrupted version of Mydoom.A that is not functional
W32.Mydoom.F@mm	February 20, 2004	a) Attempts to perform a DOS against www.microsoft.com and www.riaa.com b) Only uses email to propagate c) Infection length is 34 568 bytes d) Opens a backdoor on TCP port 1080 e) Potentially deletes files with an extension of .mdb, .doc, .xls, .sav, .jpg, .avi, and .bmp f) Terminates certain running processes
W32.Mydoom.G@mm	March 2, 2004	a) Attempts to perform a DOS against www.symantec.com b) Only uses email to propagate c) Infection length is approximately 20 000 bytes d) Opens a backdoor on TCP ports 80 and 1080 e) Potentially deletes files with an extension of .jpg, .avi, .bmp f) Terminates certain processes
W32.Mydoom.H@mm	March 3, 2004	a) Attempts to perform a DOS against www.symantec.com b) Only uses email to propagate c) Infection length is approximately 32 256 bytes d) Opens a backdoor on TCP ports 80 and 1080 e) Potentially deletes files with an extension of .jpg, .avi, .bmp f) Terminates certain running processes

W32.Mydoom.I@mm	April 15, 2004	a) Does not attempt to perform a DOS b) Only uses email to propagate c) Infection length is 44 544 bytes d) Does not install a backdoor
W32.Mydoom.J@mm	April 20, 2004	a) Does not attempt to perform a DOS b) Infection length is 50 688 bytes (.exe), approx 50 800-51 000 bytes (.zip) c) This is an encrypted virus with keylogging capabilities d) Terminates processes, including AV and security software e) Does not install a backdoor f) Written in C++ and is packed with UPX
W32.Mydoom.K@mm	May 18, 2004	a) Does not attempt to perform a DOS b) Only uses email to propagate c) Infection length is 50 176 bytes d) An encrypted virus e) Allows unauthorized remote access and terminates processes, including AV and security software

Table 3 - W32.Mydoom.A@mm Variants

2.5. Description

Mydoom is a virus, however, does have the characteristics of a worm once it is launched and can be considered a hybrid. Its only attack vector is out of date anti-virus (AV) signature files. Out of date AV signature or definition files are possible for several reasons.

a) AV software not installed
b) AV software installed but service not started
c) There is no managed AV server and it is up to the user to configure automatic updates and/or install updates manually as necessary
d) There is no disk space left on the hard drive (more common than you would think!)

The difference between a virus and a worm is that in order for a virus to be successfully executed it requires human interaction. For example, Mydoom will come with a subject line, a message and an attachment. When a user sees the attachment in their inbox or in a file downloaded from the Peer-to-Peer (P2P) file sharing service Kazaa this will not initiate the spread of the virus. The user must physically double-click or open the attachment to start its mass spread. At this

point the virus turns into a worm because it becomes self-propagating with its own built-in SMTP engine.

Mydoom is a virus that spreads by two means; email and Kazaa. When initiated via email it can come with several different subject lines, message bodies and attachments. It has its own SMTP engine and DNS client so that it does not have to rely on internal mail servers or DNS servers. Once executed it will also look for Kazaa. If Kazaa is found it will place itself in a download directory with one of several filenames and extensions. These files are now available for download by any Kazaa user (refer to "Signatures of the Attack" section). When a system is infected, Mydoom will attempt a Denial of Service (DOS) attack on http://www.sco.com. This will start on February 1, 2004 and end on February 12, 2004. Also, a backdoor is installed that listens on ports 3127-3198 and acts as a TCP Proxy [12]. "A TCP proxy is a server which acts as an intermediary between a client and another server, called the destination server. Clients establish connections to the TCP proxy server, which then establishes a connection to the destination server." (6.824 Lab 3: A TCP Proxy, p.1).

http://www.pdos.lcs.mit.edu/6.824/labs/tcpproxy.html [12]

This will allow a malicious user to connect to the backdoor and perform un-ethical acts such as attacking another infected system or forwarding spam.

2.6. Signatures of the Attack

Mydoom leaves several signatures behind once the malcode is executed. The first thing a 'happy clicker' will see when Mydoom launches is a Notepad window displaying random characters (Figure 1).

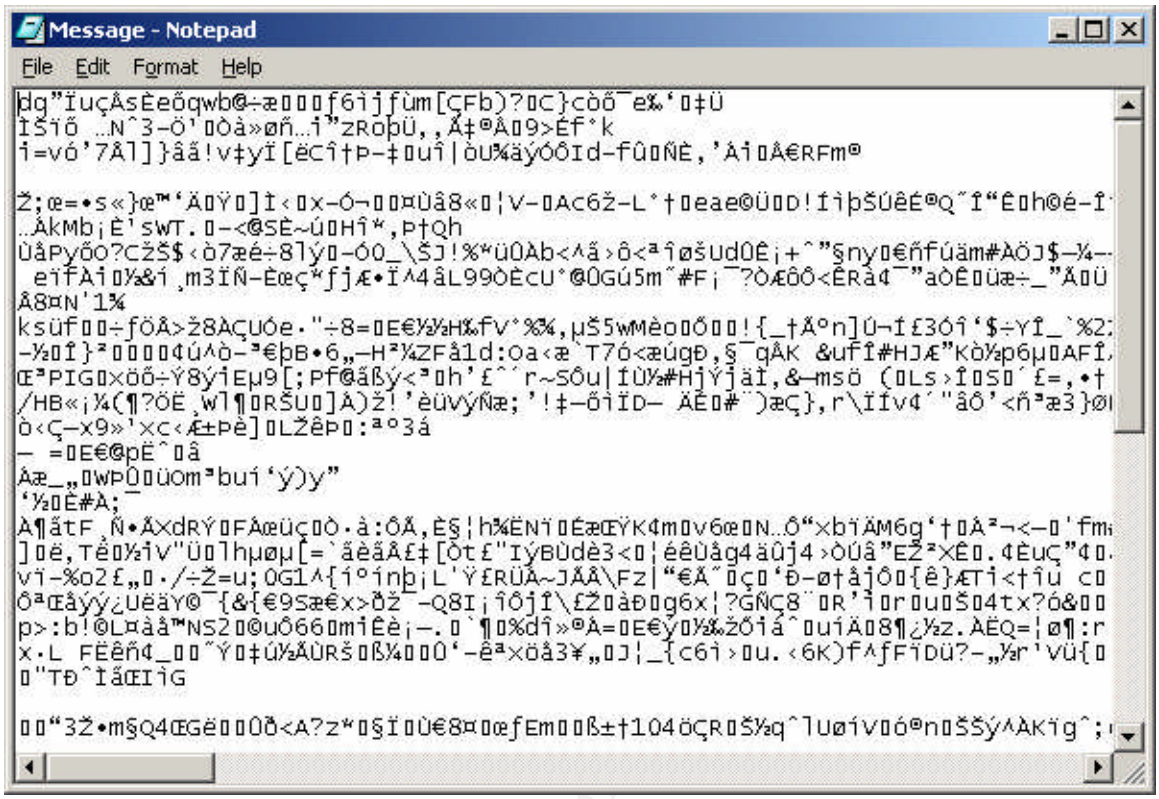

Figure 1 – Random characters displayed from Mydoom execution

The typical uneducated user will think nothing much of this window, close it and go about their business. In the meantime Mydoom will open a backdoor on the host system and look for other systems to infect.

The backdoor installed from Mydoom initially listens on port 3127 (Figure 2) and creates a file called shimgapi.dll [13].

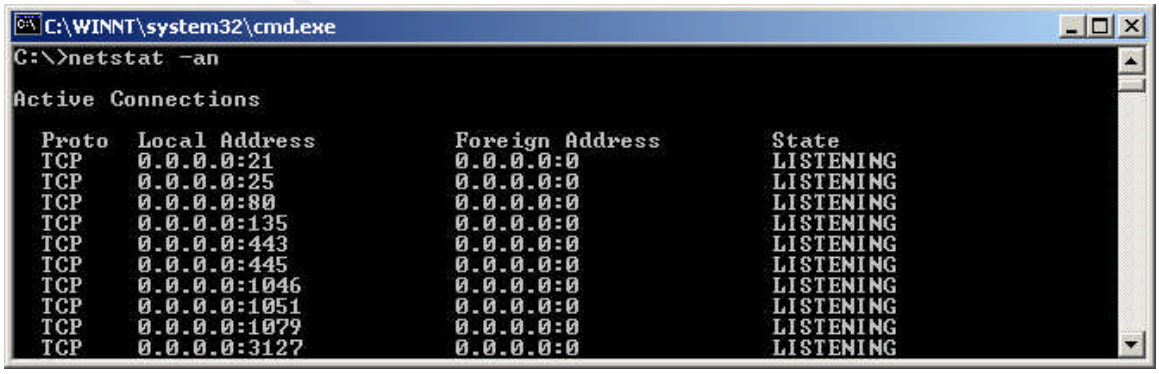

Figure 2 – netstat –an is used to display all connections and listening ports (-a) in numerical form (-n)

The last entry shows that port 3127 is active and listening.

Once a connection is made to port 3127 it will reopen on port 3128 (Figure 3) and continue to do this until port 3198. The backdoor gives the attacker the ability to perform actions such as executing arbitrary files or forwarding spam.

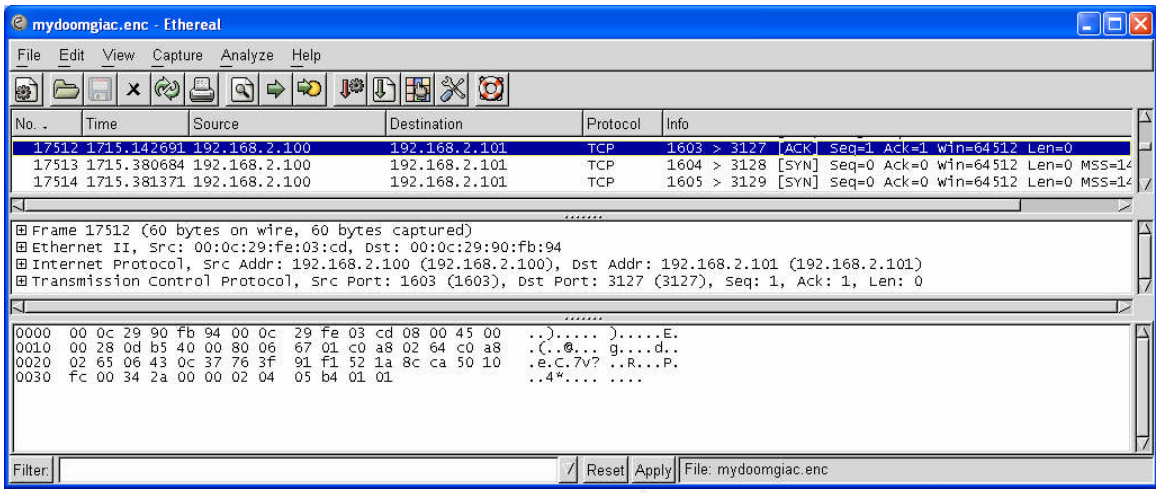

Figure 3 – Packets from an infected Mydoom system

Here is an explanation of the Ethereal packet capture above [14]:

Top Window

a) No. – packet number
b) Time – the time the packet was received relative to when the packet capture started
c) Source – IP address of where the packet came from
d) Destination – IP address of where the packet originated from (in this case the infected system)
e) Protocol – the protocol that the source and destination communicate with
f) Info – a summary of the information in the middle window (notice there is communication on port 3127)

Middle Window

a) Detailed information on the packet highlighted. This includes information on the frame, IP and TCP.

Bottom Window

a) Packet contents in hexadecimal format.

The file shimgapi.dll is added to %System% (C:\Windows\System for Windows 9x and ME, C:\WINNT\System32 for NT&2000 and C:\Windows\System32 for Windows XP) and to the registry key HKEY_CLASSES_ROOT\CLSID\{E6FB5E20-DE35-11CF-9C87-

00AA005127ED}\InProcServer32 [15] (Figure 4). Explorer.exe then loads shimgapi.dll [16].

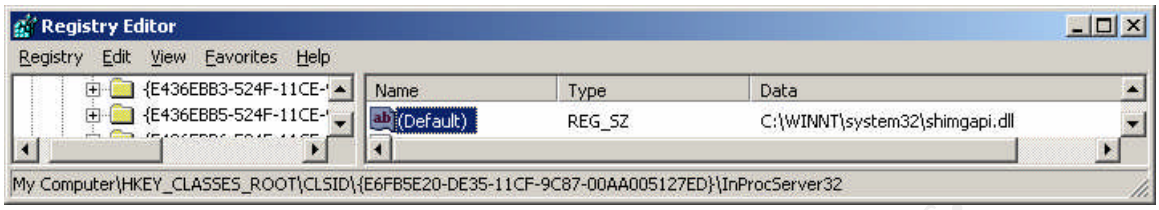

Figure 4 – Shimgapi.dll addition to registry after Mydoom infection

When executed Mydoom also copies itself as taskmon.exe to %System% (C:\Windows\System for Windows 9x and ME, C:\WINNT\System32 for NT&2000 and C:\Windows\System32 for Windows XP) [15]. The value TaskMon is added to the following registry keys [16] (Figures 5 and 6):

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
and
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run

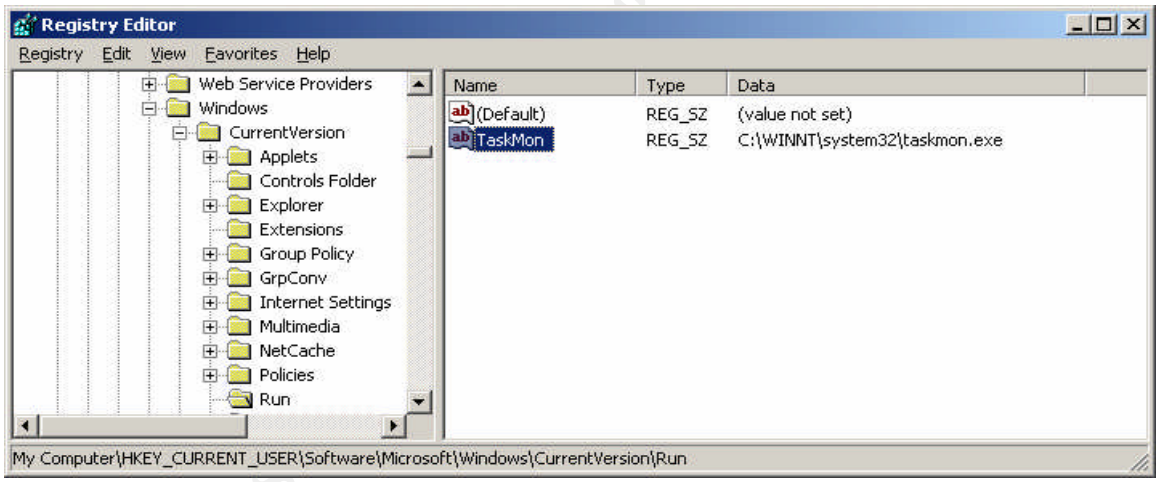

Figure 5 – Addition of TaskMon in HKEY_CURRENT_USER

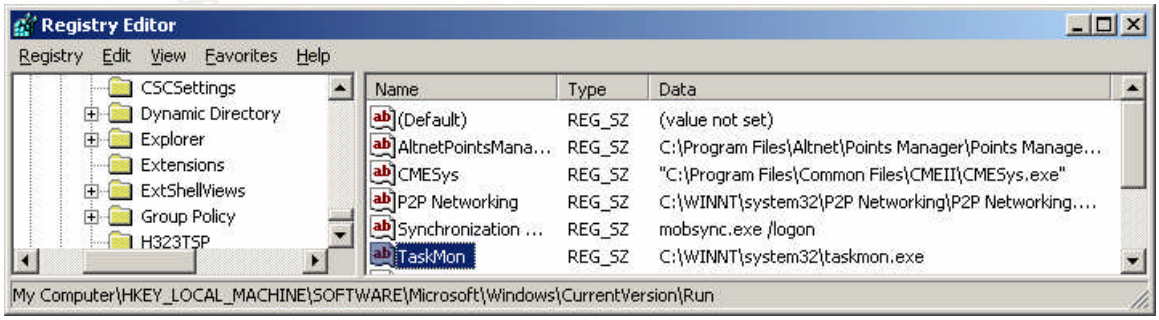

Figure 6 – Addition of TaskMon in HKEY_LOCAL_MACHINE

The final trace of a Mydoom infection is the creation of the following registry keys [17] (Figures 7 and 8):

HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\
Explorer\ComDlg32\Version
and
HKEY_LOCAL_MACHINE\ Software\Microsoft\Windows\CurrentVersion\
Explorer\ComDlg32\Version

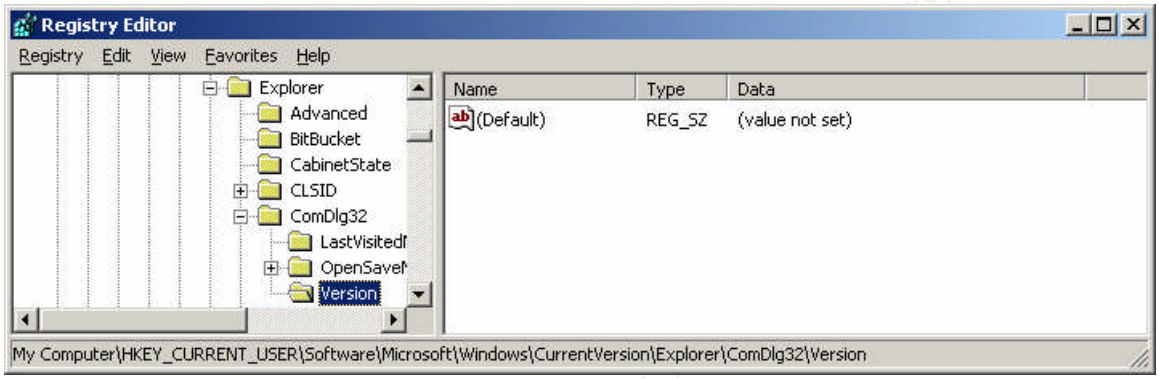

Figure 7 – Addition of registry key in HKEY_CURRENT_USER

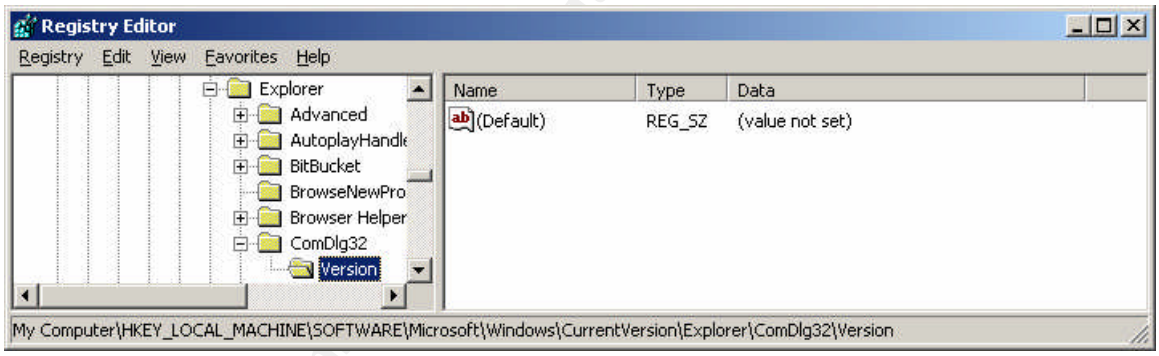

Figure 8 – Addition of registry key in HKEY_LOCAL_MACHINE

Once Mydoom has added its files, modified the registry and installed a backdoor it looks to propagate via email and Kazaa.

Mydoom will search for email addresses with many different extensions and use its own SMTP engine to spread. The email may have different subject lines, message bodies, attachments and attachment extensions. In addition, the sender's email address may be spoofed. Refer to Appendix A for this specific information [18].

If Kazaa is installed Mydoom will copy itself into the Kazaa shared folder for all P2P users to download. Mydoom uses enticing file names to try and lure the user into downloading and executing the virus [19].

- winamp5
- icq2004-final
- activation_crack
- strip-girl-2.0bdcom_patches
- rootkitXP
- office_crack
- nuke2004

Possible extensions used are:

- .bat
- .exe
- .scr
- .pif

Only AV signature files more current than those released late afternoon on January 26, 2004 will detect and prevent the spread of Mydoom should one of its attachments be opened. Refer to 'The Incident Handling Process' for more detail on what happens when AV signature files are current.

3.0 THE PLATFORMS/ENVIRONMENTS

The Viruses Unlimited network is currently in the final stages of the migration from Windows NT Workstation to Windows 2000 Professional for desktops and laptops. All systems have Symantec's Corporate Edition Anti-Virus installed (version 8.0) as well as the Microsoft Office 2003 suite. They have recently purchased an automated patch management solution and all Windows 2000 clients have the agent installed and comply with patch levels. The Windows NT Workstations remain out of compliance with respect to patches.

All servers (domain controllers, web servers, application servers) have been migrated to the Windows Server 2003 operating system. They are updated with Norton Anti-Virus (NAV) and patches. Exchange Servers have been migrated from 2000 to 2003 and are configured to block the following attachments at the gateway through Mail Marshal:

FileName=*.bat;*.chm;*.cmd;*.com;*.pif;*.hlp;*.hta;*.inf;*.ins;*.js;*.jse;*.reg;*.sct;*.shs;*.vb;*.vbe;*.vbe;*.vbs;*.wsc;*.wsf;*.wsh;*.ade;*.adp;*.bas;*.cpl;*.crt;*.isp;*.msc;*.msi;*.msp;*.mst;*.pcd;*.scr;*.lnk;*.exe;*.ceo;*.uue

The Exchange Servers also have Symantec Anti-Virus installed at the gateway and OS levels.

Firewalls in use are Cisco PIX version 6.1, routers are Cisco IOS version 12.3(1) and the core network switches are running Cisco version 12.1(14) EA1a. The network is connected through fibre.

3.1. Victim's Platform

- P4 2.6 GHz desktop running Windows 2000 Professional Service Pack 4
- 512 Mb RAM
- 30 Gb hard drive
- Symantec Corporate Edition Anti-Virus software installed – version 8.0 (updates pushed down from the AV parent server hourly assuming that there are updated definition files – current signature files dated January 25, 2004)
- The patch management agent is installed and is currently in compliance with respect to patch levels
- Complete Microsoft Office 2003 package installed
- P2P file sharing software not installed

3.2. Attacking System

This attack was carried out from inside the target network, therefore, the source network and target network are the same. Refer to Figure 9 for details. Specs of the attacking system:

- PIII 1GHz DELL Latitude C610 laptop running Windows 2000 Professional Service Pack 4
- 640 Mb RAM
- 28 Gb hard drive
- Norton Internet Security Professional 2004 installed (includes anti-virus, anti-spam, personal firewall and personal IDS). AV signatures up to date (January 25, 2004).
- The following Windows and Linux based security/hacking tools were installed:
    a) ethereal network protocol analyzer (requires Winpcap)
    b) windump network packet sniffer (requires Winpcap)
    c) mpack (encodes binary files in MIME format mail messages)
    d) netcat (can read and write data across a network)
    e) superscan port scanner
    f) nessus vulnerability scanner
    g) nmap port scanner
    h) vmware workstation (virtual operating systems)
- All security patches up to date
- Complete Microsoft Office 2003 package installed

3.3. Target Systems

- All systems on the viruses.unlimited.local network that do not have the most current anti-virus definition file installed.

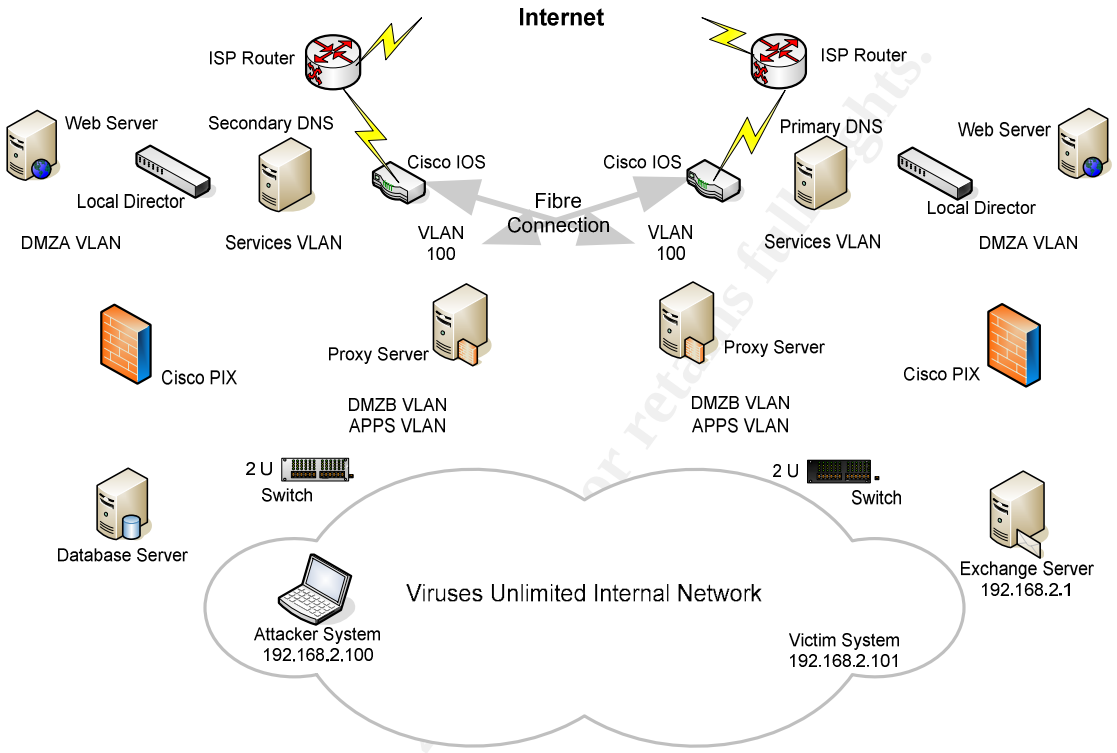

Figure 9 – Viruses Unlimited Network Diagram

4.0 STAGES OF THE ATTACK

4.1. Reconnaissance

The websites Samspade (network query tool)[20], ARIN (North American IPs registered here) [21] and Netcraft (valuable information on Web servers) [22] were initially researched to gather information on Viruses Unlimited prior to launching the attack. However, this proved to be unnecessary since it was trivial to plug into the corporate network and spread the virus from the inside.

Mr. Joe Infect competes directly in the Viruses Unlimited market space. Early in his career, he was a Systems Administrator and then promoted to a Systems Engineer before realizing he could profit the most from making sales. Therefore, he became a salesperson. Joe was a well-rounded individual who had both technical savvy as well as great soft skills. He was one of those people that

could usually get the things he wanted by looking professional and having the 'gift of the gab'. However, he tended to take it to the extreme at times when things did not work out his way. The loss of a multi-million dollar sale to his archrivals at Viruses Unlimited made Joe an unhappy individual and he sought revenge.

Joe was not quite sure how he was going to get his revenge. He only knew that he wanted to do something that would give Viruses Unlimited bad publicity as well as shut down production for an extended period so that money would be lost.

Ideally, Joe wanted to launch an attack inside of their network with a zero-day attack. A zero-day attack takes advantage of an exploit that is circulating in the wild to which there is no defence. For example, the rapid circulation of a virus in the wild that has no signature files developed to mitigate the threat, or malcode that is made publicly available to exploit vulnerabilities in un-patched OS' or applications and no vendor patch exists. Therefore, any organization hit with a zero-day attack would be adversely affected.

Joe was a member of several security newsgroups and had his favourite websites book marked so that he could keep up to speed on any new exploits circulating in the wild. At times Joe became frustrated because he thought the day would never come when a zero-day exploit would be released that he had the knowledge to take advantage of. On January 26th, 2004 he would have to wait no longer.

As usual, Joe would get into the office at 8:30am, make himself a coffee and check his email. He had several interesting emails from various subscribed mailing lists about a new virus circulating in the wild called Mydoom. The first thing Joe did was check out Symantec's website for information regarding Mydoom. Symantec had noted the threat as a category 4 (high distribution in the wild). Joe also noticed that signature files were not yet developed to mitigate the spread of this virus.

After reading Symantec's advisory and going through the rest of his emails, Joe noticed a few suspicious subject lines (with email attachments) that were the same as those described as Mydoom on Symantec's advisory page. The light bulbs turned on and he realized that those emails were the Mydoom virus. Joe safely saved those attachments to his laptop. This was his chance to cause havoc among the Viruses Unlimited network since the spread of the virus was rapid and signature files were still being developed and tested. Updated signature files was the only method of defense against this attack once a user opened the attachment (refer to the 'Incident Handling Process' section). However, since Joe could not launch the virus from the outside (assuming that external relaying was disabled and he wanted to spoof the email) he had no choice but to try and physically get inside the Viruses Unlimited network where

he assumed internal relaying would not be disabled. This is where Joe's charm and charisma came into play as his only choice was to social engineer his way inside their office building.

By 11:30am Joe Infect was in the Viruses Unlimited parking lot with laptop in hand. He took the elevator to their office on the 13th floor and buzzed to try and get in. The receptionist politely opened the door and asked how she could assist him. Joe (who was familiar with his competitor Viruses Unlimited) said that he was there for a meeting with CEO, John Doubleclick. The receptionist took his name and without hesitation let Joe in to the office and escorted him into the executive meeting room. Since Joe was early for their supposed lunch meeting he asked if there was a data port in the room so that he could check his email. The receptionist kindly assisted Joe and his wish was granted. Just like that Joe was plugged into the internal corporate network and the receptionist left the room back to her desk. He now had a valid DHCP assigned address from the internal network. Joe's clever and effortless social engineering mission was accomplished.

4.2. Scanning

The scanning phase of this attack was trivial since Joe was already on the inside and plugged in. He had some of his favourite security/hacking tools installed on his laptop that he would use to his advantage. The key piece of information that Joe required was the IP of a valid mail server(s) in order to be able to execute his plan. His plan consisted of spoofing an email to the whole of the Viruses Unlimited organization with Mydoom attached in hopes that one user would fall for his trick and open the attached malcode to initiate its propagation.

Since a mail server is required to have port 25 open Joe would use one of his port scanning tools to search for systems with this port open. The two tools he preferred were Superscan (used on Windows platforms) [23] and Nmap (mainly used on Linux platforms) [24]. In this case he was using his host OS, Windows 2000 Professional to scan and not one of his Linux virtual OS'. Therefore, Superscan was the tool of choice. Figure 10 shows the results of Joe's port scan for port 25. Besides his own system he noticed one other system that had port 25 open. Based on the host name of the system and its IP address Joe was confident that this was a server and not a workstation. He was convinced that win2003std was a Windows Server 2003 system and was also the Exchange Server. This was enough for Joe to go into the next phase of his attack.

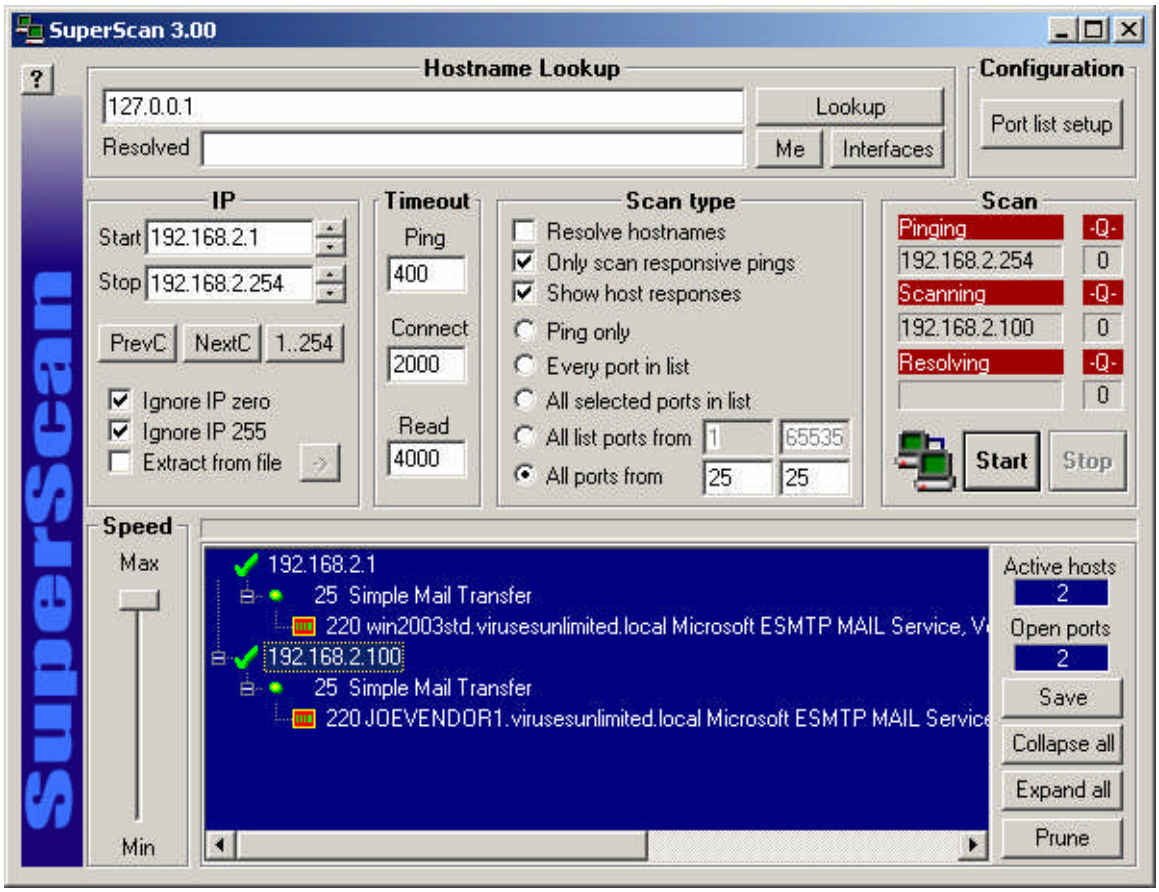

Figure 10 – SuperScan port scan results after scanning for port 25

4.3. Exploiting the System

Now that Joe was confident he had the IP address of a valid email server he would attempt to launch Mydoom internally by spoofing an email that appeared to be from the finance department on behalf of Viruses Unlimited CEO, John Doubleclick.

First, Joe typed out his email message body labeled bonus.txt (Figure 11).

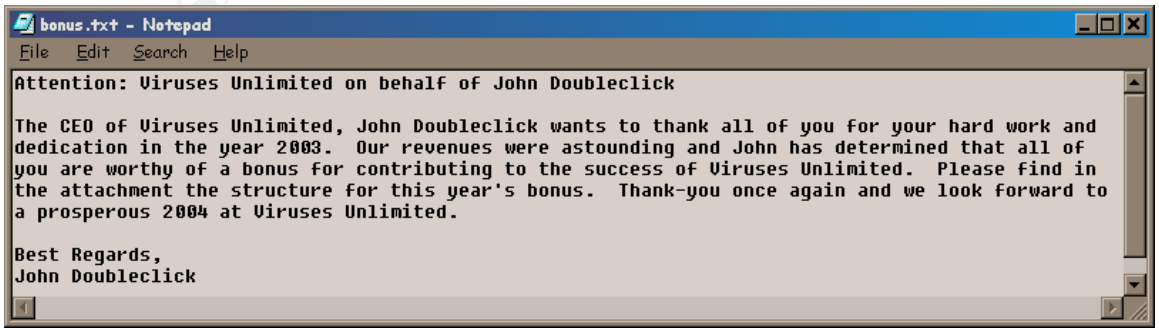

Figure 11 – Joe's fake message to Viruses Unlimited

Next, he launched the program mpack [25]. Mpack is a handy program that packs and encodes files in MIME *(*Multipurpose Internet Mail Extensions*)* format (Figure 12).

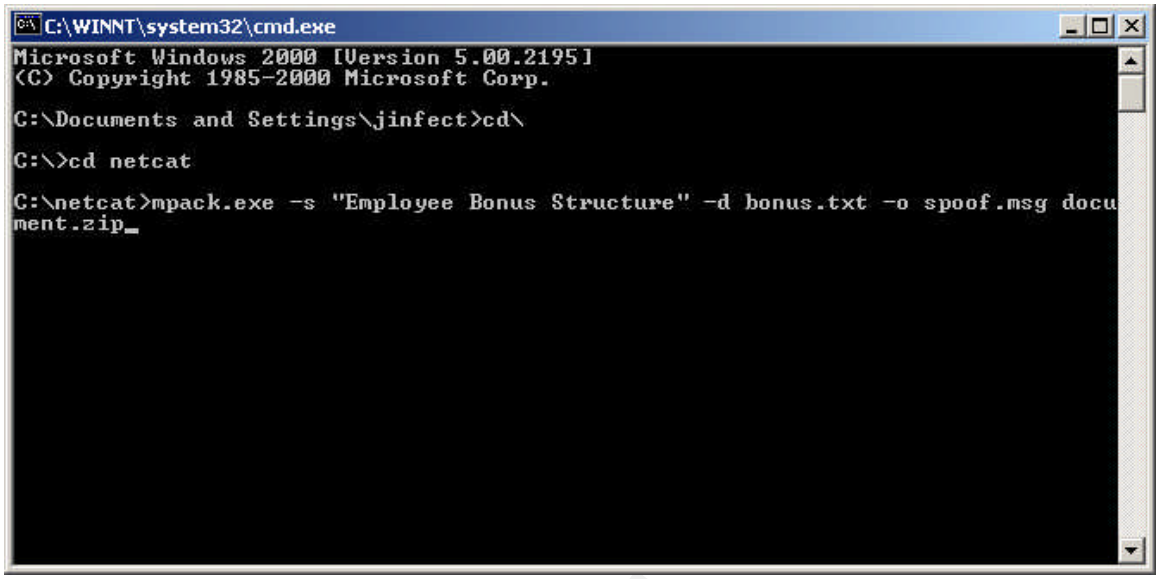

Figure 12 – Executing mpack

Joe had saved a few copies of Mydoom on his laptop. One of the viruses was labeled document.zip. This is the attachment he chose to use assuming that .zip files were not blocked internally at the mail gateway. He figured the extensions .bat, .cmd, .exe, .pif, and .scr would be blocked at the gateway as a best practice.

Explanation of the command used in Figure 12:

a) Joe changed to the directory where mpack is installed
b) mpack is launched with the command mpack.exe
c) The “-s” option is used to give the spoofed email an attractive subject line
d) The “-d” option is used as the description file, which really is the message body (in this case bonus.txt)
e) The “-o” option is used as the output file where the spoofed email is packed into (spoof.msg)
f) document.zip is the attached virus that is packed into the output file spoof.msg

Joe’s next step was to edit the output file spoof.msg to make it appear even more convincing that is was being sent by the CEO. He opened spoof.msg in notepad to do the editing (Figure 13).

helo localhost
mail from: finance@virusesunlimited.local
rcpt to: everybody@virusesunlimited.local
data
Message-ID: <3288742957@random-pc>
Mime-Version: 1.0
To: everybody@virusesunlimited.local
Subject: Employee Bonus Structure
Content-Type: multipart/mixed; boundary="-"

This is a MIME encoded message. Decode it with "munpack"
or any other MIME reading software. Mpack/munpack is available
via anonymous FTP in ftp.andrew.cmu.edu:pub/mpack/

Attention: Viruses Unlimited on behalf of John Doubleclick

The CEO of Viruses Unlimited, John Doubleclick wants to thank all of you for your hard work and dedication in the year 2003. Our revenues were astounding and John has determined that all of you are worthy of a bonus for contributing to the success of Viruses Unlimited. Please find in the attachment the structure for this year's bonus. Thank-you once again and we look forward to a prosperous 2004 at Viruses Unlimited.

Best Regards,
John Doubleclick
CEO, Viruses Unlimited

Content-Type: application/octet-stream; name="**document.zip**"
Content-Transfer-Encoding: base64
Content-Disposition: inline; filename="**bonus_structure.zip**"
Content-MD5: h4yULojTzOao98Mk/M25WQ==

0M8R4KGxGuEAAAAAAAAAAAAAAAAAAAPgADAP7/CQAGAAAAAAAAAA
AAAAABAAAAGwAAAAAAAAAEAAA/v///wAAAD+////AAAAABoAAAD/////////////
//
AgAAAOIAAABcAHAACwAASm9obiBCYWxpbnQgICAgICAgICAgICAgICAgICA
gICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICA
gICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIEIAAgCwBGEBAgAAAMAB
AAA9AQYAAQACAAMAnAACAA4AGQACAAAAEgACAAAAEwACAAAArwECA
AAAvAECAAAAPQASAGgBWgAbMO4vOAAAAAAAAQBYAkAAAgAAAI0A
AgAAACIAAgAAAA4AAgABALcBAgAAANoAAgAAADEAGgDIAAAA/3+QAQAA
AAAAAAUBQQByAGkAYQBsADEAGgDIAAAA/3+QAQAAAAAAAAUBQQByAG
kAYQBsADEAGgDIAAAA/3+QAQAAAAAAAAUBQQByAGkAYQBsADEAGgDIA
AAA/3+QAQAAAAAAAAUBQQByAGkAYQBsADEAGgDIAAIA/3+QAQAA
AAIAAAUBQQByAGkAYQBsAB4EHAAFABcAACIkIiMsIyMwXyk7XCgiJCIjLCMj
MFwpHgQhAAYAHAAAIiQiIywjIzBfKTtbUmVkXVwoIiQiIywjIzBcKR4EIgAHAB0A

ACIkIiMsIyMwLjAwXyk7XCgiJCIjLCMjMC4wMFwpHgQnAAgAIgAAIiQiIywjIzAuM
DBfKTtbUmVkXVwoIiQiIywjIzAuMDBcKR4EwAqADIAAF8oIiQiKiAjLCMjMF8pO1
8oIiQiKiBcKCMsIyMwXCk7XygiJCIqICItIl8pO18oQF8pHgQuACkAKQAAXygqIC
MsIyMwXyk7XygqIFwoIywjIzBcKTtfKCogIi0iXyk7XyhAXykeBD8ALAA6AABfKCIk
IiogIywjIzAuMDBfKTtfKCIkIiogXCgjLCMjMC4wMFwpO18oIiQiKiAiLSI/P18pO18o
QF8pHgQ2ACsAMQAAXygqICMsIyMwLjAwXyk7XygqIFwoIywjIzAuMDBcKTtfK
CogIi0iPz9fKTtfKEBfKeAAFAAAAAAA9f8gAAAAAAAAAAAAAADAIOAAFAABA
AAA9f8gAAD0AAAAAAAAAADAIOAAFAABAAAA9f8gAAD0AAAAAAAAADAIO
AAFAACAAAA9f8gAAD0AAAAAAAAADAIOAAFAACAAAA9f8gAAD0AAAAAA
AAAADAIOAAFAAAAAAA9f8gAAD0AAAAAAAAADAIOAAFAAAAAAA9f8gAA
D0AAAAAAAAAADAIOAAFAAAAAAA9f8gAAD0AAAAAAAAADAIOAAFAAAAA
AA9f8gAAD0AAAAAAAAADAIOAAFAAAAAAA9f8gAAD0AAAAAAAAADAIOA
AFAAAAAAA9f8gAAD0AAAAAAAAADAIOAAFAAAAAAA9f8gAAD0AAAAAA
AAADAIOAAFAAAAAAA9f8gAAD0AAAAAAAAADAIOAAFAAAAAAA9f8gAAD0
AAAAAAAAADAIOAAFAAAAAAA9f8gAAD0AAAAAAAAADAIOAAFAAAAAA
AAQAgAAAAAAAAAAAAADAIOAAFAABACsA9f8gAAD4AAAAAAAAADAIOA
AFAABACkA9f8gAAD4AAAAAAAAADAIOAAFAABACwA9f8gAAD4AAAAAA
AAADAIOAAFAABACoA9f8gAAD4AAAAAAAAADAIOAAFAABAAkA9f8gAAD4
AAAAAAAAADAIOAAFAAFAAAAAQAgAAAIAAAAAAAAADAIJMCBAAQgAP/
kwIEABGABv+TAgQAEoAE/5MCBAATgAf/kwIEAACAAP+TAgQAFIAF/2ABAgA
AAIUADgCjCAAAAAAGAFNoZWV0MYUADgCaEQAAAAAGAFNoZWV0MoUAD
gChEgAAAAAGAFNoZWV0M4wABAABAAEAwQEIAMEBAABgaQEA/AC//////////
//
//////////////////////////////////////9SAG8AbwB0ACAARQBuAHQAcgB5AAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAFgAFAf//
////////AgAAACAIAgAAAAAAwAAAAAAAAEYAAAAAAAAAAAAAAAAAAAAAAAAA
AAP7///8AAAAAAAAAAFcAbwByAGsAYgBvAG8AawAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAASAAIB//////
//////////AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAKgTAAAAAAAABQBTAHUAbQBtAGEAcgB5AEkAbgBmAG8AcgBtAGEAd
ABpAG8AbgAAAAAA
EkAbgBmAG8AcgBtAGEA/SEADAIOAAFAAAAAAA9f8gAAD0AAAAAAAAADA
IOAAFAAAAAAA9f8gAAD0AAAAAAAAADAIOAAFAAAAAAA9f8gAAD0AAAA
AAAAADAIOAAFAAAAAAAQAgAAAAAAAAAAAAADAIOAAFAABACsA9f8gA
AD4AAAAAAAAADAIOAAFAABACkA9f8gAAD4AAAAAAAAADAIOAAFAABA
CwA9f8gAAD4AAAAAAAAADAIOAAFAABACoA9f8gAAD4AAAAAAAAADAI
OAAFAABAAkA9f8gAAD4AAAAAAAAADAIOAAFAAFAAAAAQAgAAAIAAAAA
AAAAADAIJMCBAAQgAP/kwIEABGABv+TAgQAEoAE/5MCBAATgAf/kwIEAAC
AAP+TAgQAFIAF/2ABAgAAAIUADgCjCAAAAAAGAFNoZWV0MYUADgCaEQAI
AMEBAABgaQEA/AC//
//

.

quit

Figure 13 – Encoded MIME output file spoof.msg opened in Notepad

Explanation of Figure 13:

The highlighted text is everything that Joe added to spoof.msg to make it appear more appealing. Helo, mail from, rcpt to, data and quit are SMTP commands [4].

a) helo tells me where the email is being sent from
b) mail from tells me who the sender is - in this case the spoofed address of finance@virusesunlimited.local
c) rcpt to tells me who the recipient is - in this case everybody@virusesunlimited.local
d) data composes the message
e) To is added so that Everybody will be seen in the To field of the email
f) Content-Type is document.zip (the virus attached in mpack)
g) Content-Disposition is bonus_structure.zip**,** which is the spoofed name of the virus that Joe chose hoping that users would not be suspicious when they saw that as the attachment name and would open it
h) quit is used to close the connection

The rest of the message is how spoof.msg appears originally after mpack packed Joe's different message components together (Subject line, the message body – bonus.txt, and the attached virus document.zip – the random characters in spoof.msg are really the MIME encoded virus).

Next, Joe used a nifty program called Netcat [26] to send the edited output file spoof.msg to its intended recipients (everybody at Viruses Unlimited).

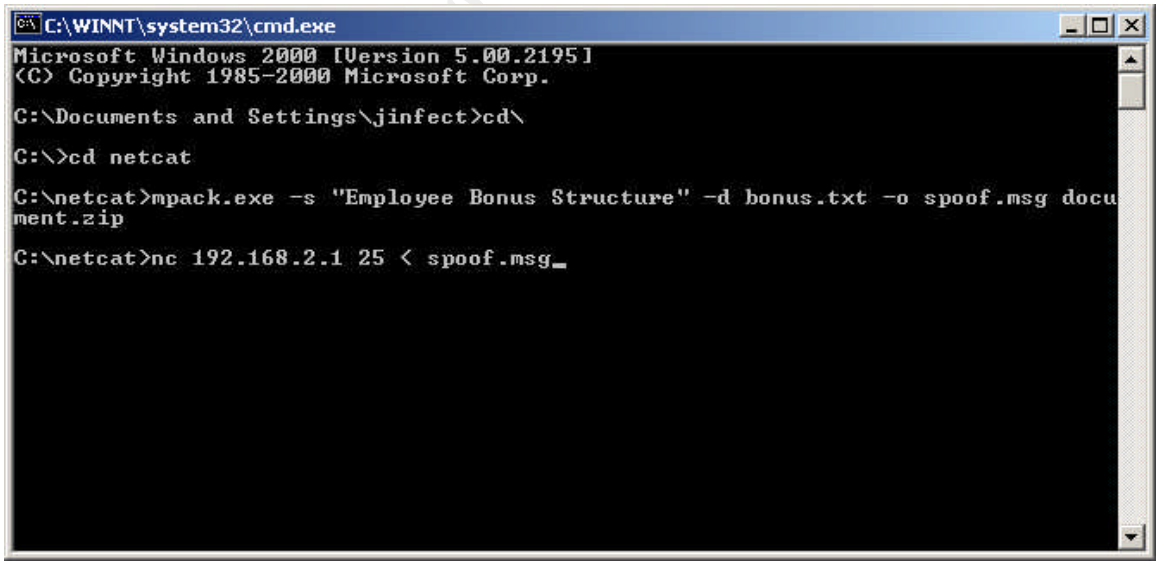

Figure 14 – Using netcat to send Joe's spoofed email

Explanation of Figure 14:

a) nc is used to start netcat
b) the IP address is that of the Exchange Server that netcat specifies as the destination system
c) 25 represents the port number used to send the message
d) spoof.msg is the file being sent

Next, Joe executes netcat to send spoof.msg (Figure 15).

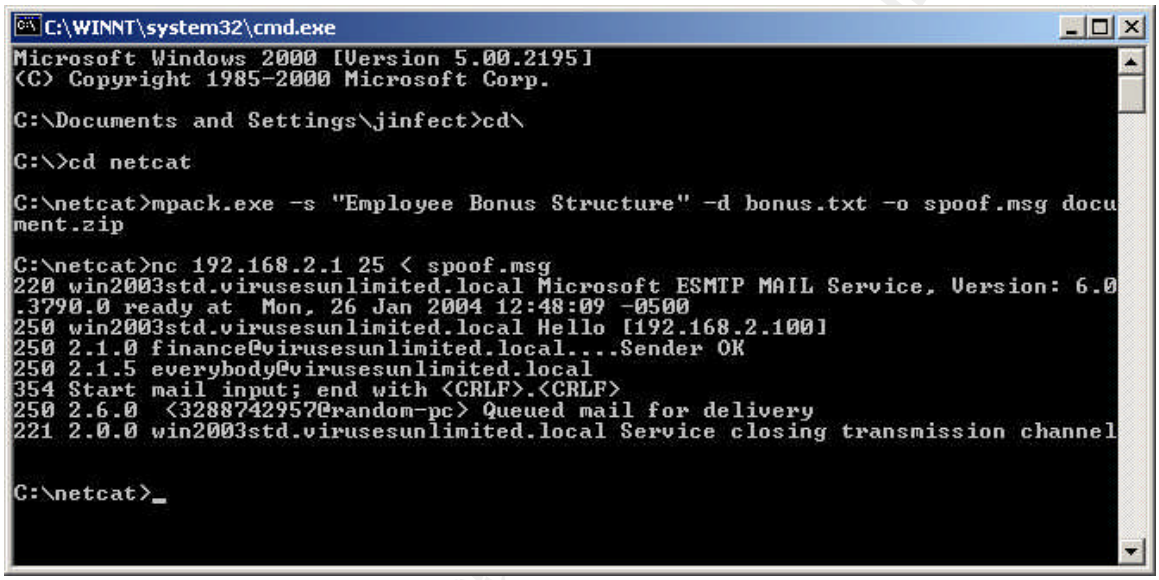

Figure 15 – Successful execution of netcat to send Mydoom

Explanation of Figure 15:

a) The first two lines tell Joe that he has made a successful connection to the mail server, the name of the mail server, version number, and time the connection was established
b) The third line tells Joe that the mail server has acknowledged the connection to his system (IP address)
c) The fourth line Joe that the sender of the message is OK
d) The fifth line tells Joe the recipient of the message
e) The sixth line starts the message input to the mail server
f) The seventh line queues the message for delivery
g) The eighth line closes the connection once the message has been successfully sent

To this point, things seemed to be going quite well for Joe and he was feeling good about himself. However, he wanted to be sure that someone had actually opened his virus-infested message and be confident that his mission was successfully accomplished.

What he did was open a command prompt on his laptop and run the packet capturing tool windump [27] to listen for traffic on port 3127 (Figure 16). Joe's goal here was to see traffic on port 3127. Since Mydoom installs a backdoor that listens on port 3127 he knew that any traffic on that port would mean at least one infected system that was propagating itself to others. Joe arrived at the Viruses Unlimited parking lot around 11:30am, he was plugged into the corporate network about 20 minutes later and it took him about another hour after that to work his evil magic before netcat was launched (12:48pm to be exact). This was perfect since most people were out for lunch and there was no one there to question Joe's presence. He continued to sit in the executive meeting room in anticipation and very anxious to see packets on port 3127. Finally, just after 1:30pm Joe's wish came true and windump was showing packets on port 3127.

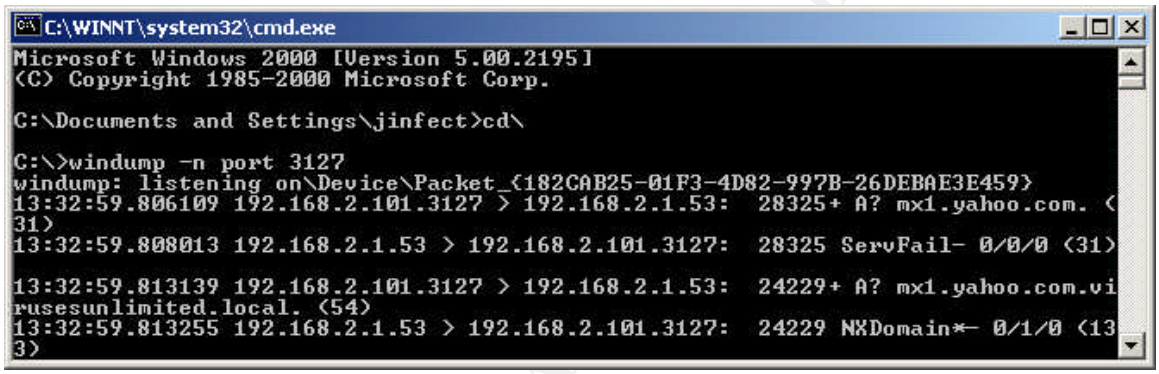

Figure 16 – windump packet captures on port 3127

Explanation of Figure 16:

a) Change to the directory where windump is installed and type windump and then your command parameters
b) –n tells windump to display IP addresses
c) port 3127 tells windump to listen only on that specific port

Explanation of the first packet Joe saw displaying port 3127 [28]:

a) 13:32:59.806109 is the timestamp
b) 192.168.2.101.3127 is the source IP address and port (the infected system)
c) 192.168.2.1.53 is the destination IP address and port (DNS)
d) 28325+ A? mx1.yahoo.com (31) represents a DNS query
    - 28325 is the DNS query number and the + indicates that it be recursive
    - A? means a query for an IP address
    - mx1.yahoo.com is the name trying to be resolved
    - (31) represents the number of bytes of data

There is a 25% chance that infected systems will attempt a DOS attack against www.sco.com beginning February 1, 2004 starting at 16:09:18 UTC and ending February 12, 2004 at 2:28:39 UTC [1]. This depends on the systems local time. Email propagation will discontinue if an infected system is part of the DOS attack, however, the backdoor will continue to listen on port 3127 even after the virus expiry date of February 12 [1].

Joe was extremely pleased with the way things had went for him in the short time span of less than two hours since he first arrived in the Viruses Unlimited parking lot. He quickly unplugged from the network, packed up his laptop and greeted the receptionist on his way out the door. She smiled back and wished him a pleasant day.

4.4. Keeping Access

Once Joe saw packets displaying port 3127 he accomplished what he had set out to do. He relied on Mydoom to keep its own access within the network as well as spread to others to give Viruses Unlimited bad publicity.

There are several ways Mydoom keeps access in a network when it is executed. Once the attachment is opened by a user it will mass mail itself with its own SMTP engine hoping that other users will do the same to continue its mass spread. Therefore, Mydoom tries to get access to as many systems as possible in order to keep access. Now let us go through how Mydoom keeps access on each system after it has been executed.

Mydoom creates the file shimgapi.dll, which acts as a proxy server and opens port 3127 to be used by attackers as a backdoor. If an attacker is successful in connecting they may potentially access network resources, execute arbitrary files or forward spam.

The file Taskmon.exe is created in the variable %System% and Mydoom copies itself there. The value 'TaskMon = %System%\taskmon.exe' is added to two different registry keys (Figures 5 and 6) so that Taskmon.exe will execute when the system starts up.

Finally, Mydoom will copy itself to the Kazaa download folder using enticing file names (if Kazaa is installed). This is a method of both keeping access and getting access. Mydoom keeps access because it stays on the infected system masquerading as a Kazza file. It may get access to other systems if someone was to download the infected file and execute it on their own system.

4.5. Covering Tracks

Joe was convinced that he would never be caught with his malicious act. After all, he was in the office for less than 2 hours and he launched the attack from

inside the network. From the time he sent the spoofed email to the time he saw packets listening on port 3127 less than 45 minutes had elapsed. As soon as he knew Mydoom was spreading internally he unplugged from the network and the DHCP address associated to his laptop was released. As far as he was concerned it was as if he was never there.

Mydoom tries to cover its tracks three ways.

a) The file %Temp%\Message is displayed as random characters in Notepad when Mydoom is executed (Figure 1). This file is deleted as soon as Mydoom is launched.
b) There are two registry keys created (Figures 7 and 8) to ensure that the random character Notepad message is only displayed once. Therefore, if a 'happy clicker' decides to re-infect themselves the Notepad window will not be displayed and they will have no hints about something suspicious happening.
c) The file Taskmon.exe that is created during execution is created in %System%. The legitimate Taskmon.exe file is created in %Windir%. Therefore, Mydoom tries to hide itself in another folder and the legitimate Taskmon.exe may be deleted.

5.0 THE INCIDENT HANDLING PROCESS

5.1. Preparation

Viruses Unlimited has been the victim of many incidents in recent years. These incidents were not limited to virus and worm outbreaks, but also attacks from compromised systems that were using VPN to connect to the internal network, a fire caused by something so silly as a person working late and forgetting about their popcorn in the microwave, and who could forget about the great blackout on August 14th, 2003 affecting major cities in the U.S. and Canada. Yes, the Incident Handlers at Viruses Unlimited have battled through all of these. From each one of those there were lessons learned to help them better prepare for the next time they were faced with an incident.

Not only did the Incident Handlers learn valuable lessons, but also so did senior management. After network outage due to Slammer, Blaster, Welchia and the blackout, there were finally executive meetings to discuss budget for things such as training, user awareness and different types of tools to aid the incident handlers in being proactive or at least help them identify the source of the problem quicker. Executive management also set out a strict mandate that security policies and processes be written and enforced. This was not the cure for cancer but it was definitely a step in the right direction for helping secure the corporate network.

The Incident Handling team consists of four members who are stationed at headquarters but on call 24x7. Although, they each hold different titles within the security group, these are the four who come together to get through the six steps of the Incident Handling process. One of them is a full-time employee at Viruses Unlimited and the other three are contractors outsourced for their technical expertise. The breakdown is as follows:

Mary Fary – Manager of Incident Handling and Operations
Brent Milkie – Manager of Security Architecture and Forensics Investigations
Kelly Grolsch – Senior Security Analyst and Application Security Specialist
Kristof Kromski – Vulnerability Specialist

Many other important 'players' from the organization (i.e. Network, Desktop, and Server team members) are pulled into the war room or call in via conference line to contribute to identifying, containing, eradicating and recovering from the incident. The Viruses Unlimited network crosses oceans with offices in Sydney, Tokyo, Hong Kong and London. They also have a major presence in Canada and the U.S.

The Incident Handling team has collectively put together a draft for the Viruses Unlimited Incident Handling process as well as a Vulnerability Management process. Both are being reviewed by Senior Management for sign-off and are outlined below:

Incident Handling Executive Summary

Ideally, the best way to "handle" an incident is to prevent it from occurring to begin with. In an ongoing effort to reduce incidents, vulnerabilities and potential risk exposures, Security Solutions is responsible for proactively monitoring and analyzing daily reports for anomaly type of activities or events. Ensuring that potential threats are uncovered and identified as an incident is vital to establishing the best course of action to minimize exposure, damage and propagation.

Security Solutions, Operations

Viruses Unlimited acknowledges that the protection of information is one of its most fundamental responsibilities. Risks to information and information resources must be managed and as such procedures, policies and guidelines must be put into practice and complied with.

The focus of this document is to outline the formal process for the daily analysis of security alerts. Intrusion Detection Sensors (Network and Host) as well as different vulnerability scans that report anomalies will be examined daily.

Roles and Responsibilities

The role of the Security Solutions Operations team is to inform and provide guidelines for information Security within the Viruses Unlimited environment. The Security team responsibilities encompass many different areas such as *Incident Management*, *Monitoring and Reporting*, *Alerts/Communications*, *Exceptions Forum or Risk Acceptance*, *Certificate Management* and *Website Reporting*.

Incident Response Management

An Incident Response Plan is required in order to bring needed resources together in an organized manner to deal with an adverse incident related to the safety and security of Viruses Unlimited resources. The incident may be a malicious code attack, unauthorized access to Viruses Unlimited systems, unauthorized utilization of Viruses Unlimited services, denial of service attacks, general misuse of systems, or hoaxes.

Incident Response

There are six stages of response [29]:

1. Preparation – one of the most important facilities to a response plan is to know how to use it once it is in place. Knowing how to respond to an incident BEFORE (*or shortly thereafter*) it occurs can save valuable time and effort in the long run.

2. Identification – identify whether or not an incident has occurred. If one has occurred, the VIRUSES UNLIMITED CSIRT can take the appropriate actions.

3. Containment – involves limiting the scope and magnitude of an incident. Because so many incidents observed currently involve malicious code, incidents can spread rapidly. This can cause massive destruction and loss of information. As soon as an incident is recognized, immediately begin working on containment.

4. Eradication – removing the cause of the incident can be a difficult process. It can involve virus removal, conviction of perpetrators, or dismissing employees.

5. Recovery – restoring a system to its normal business status is essential. Once a restore has been performed, it is also important to verify that the restore operation was successful and that the system is back to its normal condition.

6. Lessons Learned – some incidents require considerable time and effort. It is little wonder that once the incident appears to be ended there is little interest in devoting any more effort to the incident. Performing follow-up activity is, however, one of the most critical activities in the response procedure. This follow-up can support any efforts to prosecute those who have broken the law. This includes changing any company policies that may need to be narrowed down or be changed altogether.

Virus Incident

The Virus Protection Task Force reviewed all of the procedures and practices currently implemented in each of the different business areas within the Viruses Unlimited Organization. The results are a number of policies that have been *published* and should be implemented enterprise wide.

Prevention/Mitigation

The activities outlined below are performed on a daily basis or as required in the case of a vulnerability issue:

- Check virus information sites (*Symantec, TrendMicro, Microsoft, CERT, etc.*) for updates and new potential threats.

- Review and recommend the necessary vulnerability patches.
    - Notify Viruses Unlimited Research and Development area of required testing and deployment.
    - Notify regional teams of vulnerability and instruct them to perform testing and provide their deployment schedule.
    - Report back to management that the alert has been sent to the necessary resources within VIRUSES UNLIMITED.

- Notify and assemble the Viruses Unlimited CSIRT when a potential virus threat or vulnerability has been discovered.

- Manage the process for anti-virus signature file updates on the desktops when a threat has been identified.

Managing A Threat - VIRUSES UNLIMITED CSIRT

Additional tasks which must be performed when a threat enters the VIRUSES UNLIMITED organization:

- Advise the VIRUSES UNLIMITED Computer Security Incident Response Team (CSIRT) of the potential threat.
- Assemble the VIRUSES UNLIMITED CSIRT to determine the best course of action to eradicate and minimize any potential damage.
- Open an Incident Handling Report.
- Liaise with the enterprise (*CSIRT*).
- Determine and act if a “Staff Alert” communication is required.
- Determine and act if the Extended Team should be notified (*HR, Compliance, Audit, Legal, etc.*).

CSIRT

A VIRUSES UNLIMITED *CSIRT* was established to address all VIRUSES UNLIMITED related incidents. The group responds to incidents such as Virus attacks, DOS attacks, Intrusion Detection Alerts, blackouts, fires, etc. Establishing a "cooperative" group allows the organization to respond quicker to any incident thus minimizing the overall exposure.

Organization

To adequately respond to an intrusion or incident, predetermined virtual teams will participate depending on the incident characteristics. As the situation develops and the impact becomes more significant, the various teams will be called to contribute.

Monitoring - Daily Activities

Vulnerability Scan Daily Reports

On all external facing hosts Nessus is used to scan for vulnerabilities.

Daily Incident Response Team (DIRT) Meetings

The intent of the meeting is to review all relevant security events over the past 24 hours, which have taken place within the Viruses Unlimited organization and assign tasks if applicable. The meeting is not a forum for resolving issues. DIRT case numbers or problem management records are assigned to ensure that the issues are tracked and resolved in a timely matter. A daily report is distributed containing the gathered data.

The topics discussed are as follows:

- Intrusion Detection Sensors (IDS)
- Anti-virus logs
- Nessus results
- Daily threat analysis (vulnerabilities etc.)

Daily Business Continuity - Daily Weather Reports\Business Disruption

The Security Solutions Operations group is responsible for Security and BRP Incident Management for Viruses Unlimited, *globally*, and as such monitors news media for weather reports and events that may cause business disruptions where Viruses Unlimited offices are located.

Examples of events that may threaten or disrupt the continued daily business functions are transit strikes, organized protests or inclement weather. A communication is sent to the target office informing them of the situation and requesting their BRP preparation information (i.e. alternative routes, remote access, etc.).

Symantec DeepSight Daily Report

DeepSight is a service provided by Symantec. The intelligence report is delivered and reviewed daily for any potential threats that are in the wild.

Other categories which are being monitored for anomalies:

Security

- Admin Accounts
- Failed Admin Logons
- IIS Hack Detection
- AV scans (looking for systems without AV and the current signature file)
- Vulnerability scans (looking for missing patches, spyware, etc.)
- Many newsgroups and mailing lists are followed for the latest vulnerability information (BUGTRAQ, NTBUGTRAQ, SECURITY FOCUS, FULL DISCLOSURE, INCIDENTS.ORG, PATCHMANAGEMENT.ORG)

Alert Communication

In the event of a Virus related incident the communication process is as follows:

- Send VIRUSES UNLIMITED CSIRT an email informing them to congregate in the 14th Floor Meeting room

- VIRUSES UNLIMITED CSIRT MUST congregate in the 14th Floor Meeting room (phone call to each member if necessary).

- Send an email to the Upper Management Group informing them that a virus issue has occurred and that the VIRUSES UNLIMITED CSIRT is assembling to address the issue. Provide the conference line information so that they can dial in for an update.

- Set-up a conference line in the war room:

  - 123-456-1179 (maximum 16 lines at the designated time)
  - followed by the # key : 1224567#
  - Subscriber password - XXXX

- Take notes at the time the events occur, who is doing what at what time - manage events - ensure not to leave the conference room. Conference call must be attended at all times.

- Send an updated email to the Upper Management Group when a milestone has been hit (i.e. virus located, virus cleaned, etc.).

Reporting - Weekly

All the results from the Security Management Reporting tool will be analyzed by Security Solutions. The findings will then be escalated to the appropriate platform Administrator and a deadline date for mitigation will be assigned.

Weekly "SnapShot"

A report is produced on a weekly basis containing all of the events that occurred for the week. The report is distributed to upper management.

The Security Categories on the report are as follows:

- Incidents
- Alerts
- Certificate Requests
- Nessus Vulnerabilities *(daily reports run against live hosts - POP)*
- Reporting Weekly Virus Activity

Viruses Unlimited - Teams

In the event of a BRP or other Security Incident, the Teams listed below must be contacted and informed of the situation.

(a) Local Technical Teams

- Intel
- UNIX
- SQL
- IIS
- Exchange
- Desktop Mangers

(b) Regional Contacts

(c) BRP IT Group

(d) HR Contact

Non Viruses Unlimited - Contacts

In the event of a BRP Incident, the people listed below must be contacted and informed of the situation.

(e) BRP Viruses Unlimited Contacts

- Bob Roy - Viruses Unlimited - Bus. Continuity Mgmt – 123-456-7889
- Lou Lones- Viruses Unlimited - Senior Manager - Business Continuity Group – 123-456-9987
- Robert Smith- Viruses Unlimited - Senior Analyst VIRUSES UNLIMITED-BRP Department – 123-456-8743

Configuring, Monitoring and Reporting - IDS

Viruses Unlimited is in the process of evaluating products, which provide comprehensive Security Management and Correlated Reporting (policy enforcement, vulnerability alerts, etc.). The evaluation will also focus on recommending a real-time host-based Intrusion Detection tool that will be managed within Security Solutions, Operations.

Administration

Intrusion Detection Systems monitor networks and systems and generate alerts based on signatures of known attack or patterns that may indicate an attack. Viruses Unlimited will be responsible for deploying, configuring, monitoring and reporting of any anomalies (known patterns that may indicate an attack). In order to leverage the risk of using one company for both Network and Host-based IDS, Viruses Unlimited has opted to use solutions from a variety of companies.

Vulnerability Management Process

Introduction

Effective vulnerability management is regarded as the ability to manage known and future vulnerabilities in an enterprise network. The ideal situation is to have a process that is able to: identify new and existing vulnerabilities, relate them to critical business assets, provide remediation solutions, and offer real-time reporting on the state of the security posture.
Cyber incidents and vulnerability numbers are on the rise along with the sophistication of attacks. This translates into a rise in the probability of attacks and infections being observed by security managers.

The most effective method of preventing these attacks is to identify and manage the vulnerabilities before they are used to compromise systems and or destroy information assets. Sound vulnerability management is the solution for this ever-growing problem. Viruses Unlimited has purchased a patch management solution and is in the implementation phase of the project. This document outlines the Viruses Unlimited process for vulnerability management and patch deployment.

Scanning Tools

There are many tools currently on the market, both open-source and shrink-wrap, which can identify vulnerable systems. Viruses Unlimited currently uses Nessus, nmap and CIS along with a number of custom applications such as perl scripts.

Identification – Systems

Security Solutions staff manages the following processes:

1. Asset Discovery - Scan IP address ranges - discover systems on VIRUSES UNLIMITED Network that are either managed or unmanaged "rogue" systems.
2. Asset Management Ownership - Identify who is responsible for each system connected to the Network.

Vulnerability Identification

Security Solutions staff research, identify, validate, categorise and communicate potential security vulnerabilities to the Viruses Unlimited Technical Management teams for testing and deployment in the Viruses Unlimited Production environment.

1. Research/Asset Identification - On a daily basis, a Vulnerability Security Specialist reviews published vulnerabilities from a number of reliable sources (Symantec, Bugtraq, etc.).
2. Identify/Analyse Environment - Vulnerabilities are identified in terms of a potential threat to the Viruses Unlimited environment.
3. Validate - A Vulnerability Security Specialist works diligently to validate the vulnerability to ensure the accuracy of the claim, assess the associated business risk, and identifies actions necessary for remediation.
4. Categorise Risk Severity - A Security Services Analyst performs an overall risk assessment should this vulnerability be exploited in the Viruses Unlimited environment. The categories are as such:

High	• Patch must be applied within a maximum of 7 days of release notice
Medium	• Patch must be applied within a maximum of 30 days of release notice
Low	• Patch must be applied within a maximum of 60 days of release notice or the normal maintenance cycle, which ever comes first.

5. Communicate - The Vulnerability Security Specialist communicates the vulnerability, globally, to the appropriate areas concerned for patch testing and deployment. The communication includes technical details along with the associated Risk Category.

Patch management

Each Technical Management area is responsible for the testing and deployment of patches and workarounds for their respective support areas i.e. UNIX Engineering, Intel, SQL, IIS, Exchange/Mail and Desktop.

1. Action Plan - Arrange for the patch or workaround to be tested on systems that you support (utilise R&D area department if applicable). If necessary, contact the application developers and inform them of your deployment plans and ensure and manage their testing progress.
2. Rollback Plan - Prior to patch deployment in the live Production environment; ensure that you have a workable (tested) rollback plan as a precautionary measure.
3. Deployment Plan - Create a prioritised deployment plan once testing has been completed to the Technical Managers satisfaction. Provide your systems administrators or a desktop resource with your prioritisation of tasks to ensure that time is spent on the most critical vulnerabilities first.

4. Deadline - If you are not able to meet the deadline outlined in the alert notification or for whatever reason the patch or workaround cannot be deployed contact the Security Solutions Incident Manager immediately. Security Solutions will work with you to ensure that an alternate solution is found to minimise the risk.
5. Accountability - Each Technical Team Manager is responsible for managing the action plan, rollback plan, deployment plan and ensuring that the patch deployment deadline is met. As such the Team Manager is responsible for providing a status of the deployment plan.

   Per Server\System Results:

   - ❖ Completed
   - ❖ More testing required
   - ❖ Non-deployment due to application not functioning properly

Scanning - Audit – Reporting

A scan will be performed weekly by Security Solutions on all systems attached to the Viruses Unlimited Network.

1. Scanning - A weekly scan will be performed by the Vulnerability Security Specialist to determine patch levels of each system.
2. Security Audit/Assessment - Security Services will create a security audit work plan and generate security audit reports correlating vulnerabilities and work history status.
3. Reporting/ Measurable Security Posture - The Vulnerability Security Specialist will maintain an ongoing status report of the patch deployment progress. Pre-defined reports will be provided to the Technical Managers.

Management Reporting

A report will be produced on a monthly basis, which will be distributed to the Viruses Unlimited Upper Management team as an assessment of the security posture.

The report will contain the following information as well as a "stop light" categorization of the patch level per system:

High	• Patch must be applied within a maximum of 7 days of release notice
Medium	• Patch must be applied within a maximum of 30 days of release notice

Low	• Patch must be applied within a maximum of 60 days of release notice or the normal maintenance cycle, which ever comes first.

Corporate policy states that the Tier 1 applications that must be installed and running on all corporate systems is Anti-Virus, the patch management agent and SMS. In addition, a security pass card is required to physically access the office space. Otherwise, a temporary security card is issued and the visitor MUST be escorted throughout the building at all times.

Viruses Unlimited Security Policies

All in scope devices shall adhere to the same policy. This policy is meant to establish a baseline for all existing configuration policies.

1. The most current security patch level versions must be applied to all in scope systems.

2. All systems must have the supported and up-to-date anti-virus software installed and configured to operate in active scanning mode.

3. All systems must have an automated method of obtaining/receiving the virus signature files.

4. The process for the distribution of new anti-virus signatures must be sound and efficient.

5. Only Viruses Unlimited sanctioned devices can be connected to the enterprise.

6. Non-essential services must be disabled.

7. All changes to WEB sites that are externally facing must be approved by Security and penetration tested.

8. All servers must have security monitoring established.

9. Only Viruses Unlimited Approved mail products will be deployed.

10. All mail servers must have an attachment blocking mechanism to block the following attachments:

FileName=*.bat;*.chm;*.cmd;*.com;*.pif;*.hlp;*.hta;*.inf;*.ins;*.js;*.jse;*.reg;*.sct;*.shs;*.vb;*.vbe;*.vbe;*.vbs;*.wsc;*.wsf;*.wsh;*.ade;*.adp;*.bas;*.cpl;*.crt;*.isp;*.msc;*.msi;*.msp;*.mst;*.pcd;*.scr;*.lnk;*.exe;*.ceo;*.uue

11. All mail servers must have content filters.

12. Multi-tier, multi-product anti-virus protection for the various technologies (best of breed) must be established.

13. A risk assessment based communication methodology for virus notification must be utilized and a CSIRT formal notification must be triggered when any part of the enterprise is compromised.

14. Infected devices must not be returned to the enterprise until certified by the CSIRT.

15. All ingress points are not to be configured for auto answer unless fire walled and/or secured with an approved access control mechanism.

16. Regular reviews must be performed to confirm that the above policies have been met with exceptions reported to CSIRT for consideration.

17. Set-up a Proxy server and restrict Internet access to Domain accounts only. For example, *Non-Domain accounts:* Developers, including contract developers bring their laptops in, connect to the Viruses Unlimited Network and start working. *Contractor Laptops:* The Support Teams do not have any control over these Laptops, which in many cases do not have any virus software, do not have the most recent patches and do not have SMS installed. Therefore it is virtually impossible to know that they are even on the network without port scanning.

18. NEVER open any files or macros attached to an email from an unknown, suspicious or untrustworthy source. Delete these attachments immediately, then "double delete" them by emptying your Recycling bin.

19. Employees working for vendors and other third party companies should not connect any computing equipment to the VIRUSES UNLIMITED corporate network without first consulting the security group. VIRUSES UNLIMITED can give guidance on the essential computer security practices needed to work in end user computing.

20. Individuals who are not Viruses Unlimited employees must have an authorized Viruses Unlimited Management sponsor to obtain a Logon ID.

21. Individuals who are not Viruses Unlimited employees must be issued a security pass card and a visitor's badge to enter the premises. When on the premises they must be escorted at all times.

22. All connections between the Viruses Unlimited internal networks and the Internet (or any other publicly accessible computer network) must include an approved firewall and related access controls.

23. Vendor access must be restricted to only the resources, which are necessary for the vendor or agent to provide the contracted product or service. Extreme caution must be taken to avoid unwarranted vendor access to information on any of the Viruses Unlimited customers or employees.

24. The appropriate forms must be completed and management approval must be obtained before accessing the Internet through Viruses Unlimited systems (computers, networks etc.).

25. Scan all file attachments for computer viruses and other destructive programs.

26. Be cautious about opening e-mail, if you are unsure of the sender. Do not open any attachments without scanning them for viruses.

5.2. Identification

The Incident Handling team was well aware of Mydoom spreading across the globe on the morning of January 26th. Part of Kristof Kromski's daily duties is to follow several newsgroups and mailing lists, as well as check security websites for new Internet threats in the wild. Kristof was also aware that there were no signature files to mitigate the virus. He was constantly checking the Symantec site for updated signature files to push out to his clients. There was a beta definition file ready for the public early afternoon, however, Kristof did not trust beta definition files after he downloaded a beta definition file earlier in the year that happened to be corrupt, which caused a lot more grief than good.

The first sign of something suspicious was shortly after 1pm. Kristof noticed there was an email sent to the 'everyone' distribution list at Viruses Unlimited. The email originated from the Finances Department and was on behalf Viruses Unlimited CEO, John Doubleclick. It was regarding employee bonuses. Kristof was not sure what to make of it and nor were the other members of the Incident Handling team. They quickly met and all found it odd, especially Kristof, Brent and Kelly since they were the contractors and would not be included in a full-time employee bonus plan. Of course these well-educated Handlers did not foolishly click on the attachment that came with the message. They decided it was best if Mary call the head of the Finances Department to see if this was legitimate. It

was now around 1:30pm. Mary asked the head of Finances (Samantha) about the email and why it was sent to everyone. Of course Samantha had no clue what she was talking about. Mary advised Samantha to check her email for something that was sent at 12:48pm from the Finances department regarding an employee bonus structure. Mary also stressed not to open the attachment. Samantha checked and sure enough she had a copy too. She hung up with Mary to call the other members of her department to see if they had sent the message. Once she had confirmed that no one from her team had sent this message she called Mary back to let her know. Mary communicated this back to the rest of the Incident Handling team around 2pm. The initial thought was a spoofed email that was potentially malicious. Mary sent out an email with high importance to the whole company telling them not to open the attachment in this message since it was not a legitimate email.

By the time CEO, John Doubleclick returned from lunch and read Mary's email; he had already opened the attachment sent by the Finances Department. He always checked his emails in chronological order. Of course when he saw the email from the Finances Department regarding employee bonuses he was too curious not to open the attachment. Figure 17 shows the message that John saw in his inbox.

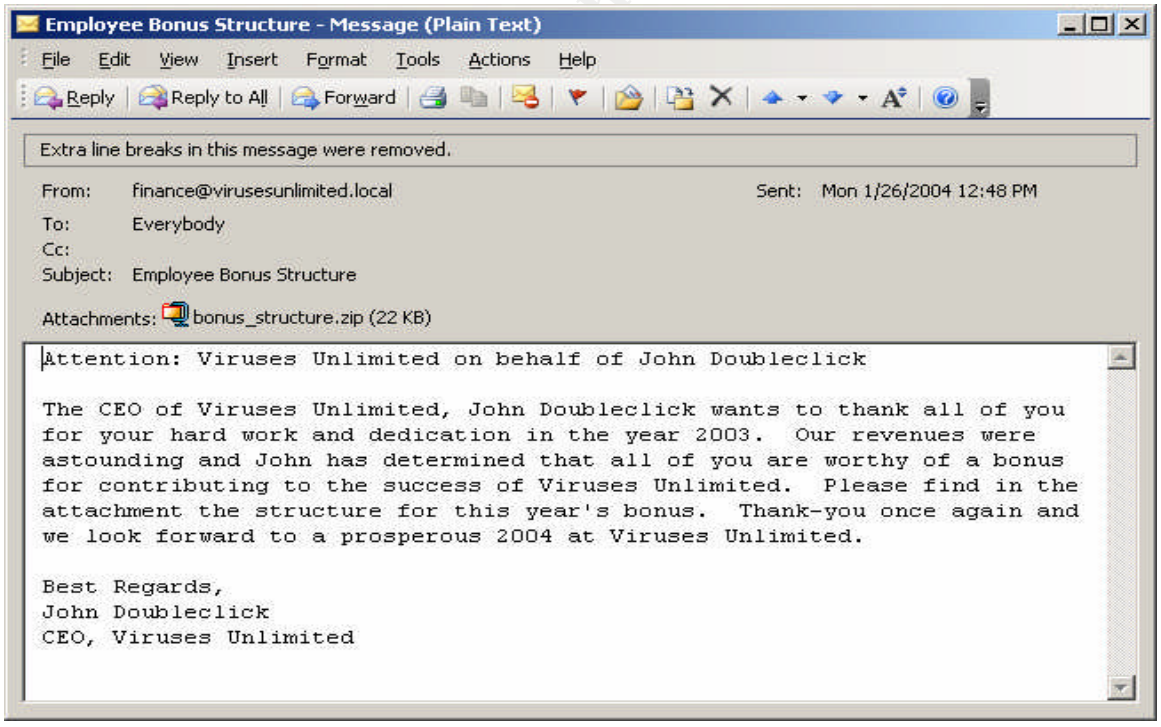

Figure 17 – Message from the Finance Department on behalf of John Doubleclick

After the attachment was opened, John furiously called his CFO to see what this was all about. John saw nothing but a text message pop-up with random

characters (Figure 1) when he opened the attachment. He explained all of this to his CFO, who in turn told John that the email was not sent by anyone at the Finances Department. At this point it was already too late. By not following company protocol John had executed Mydoom within his own organization.

As soon as John read Mary's email he called her to let her know he had opened the attachment and wanted her to check his PC for any anomalies. She assigned Kristof to the task. John only sat three floors below so Kristof was at his desk in a few minutes. The first thing he did was run a full scan checking for viruses (signature files were from January 25th, 2004). The PC turned up clean. He then checked the registry to make sure all critical patches were applied and they were. While the scan was running Kristof asked John what exactly happened when he opened the attachment. John stated that a text window popped up for a second or two that was filled with a bunch of random characters. At that point Kristof was certain that John had been infected with Mydoom since the pop-up text window was a characteristic of the virus. He printed out a copy of the Mydoom advisory from Symantec's site so that he could check to see if John's PC had other traces that Mydoom leaves behind. Kristof looked for the files shimgapi.dll (Figure 18), taskmon.exe in %System% (Figure 19) and also checked to see if any port from 3127 to 3198 was open (Figure 20). If all of the above were true then Kristof could confirm a Mydoom infection.

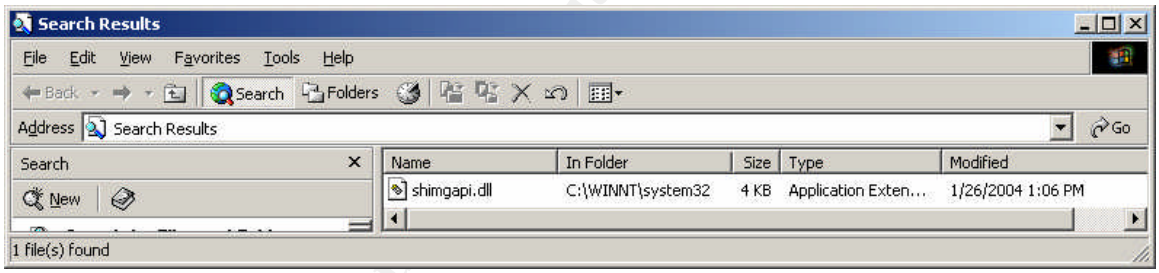

Figure 18 – Search results for the file shimgapi.dll

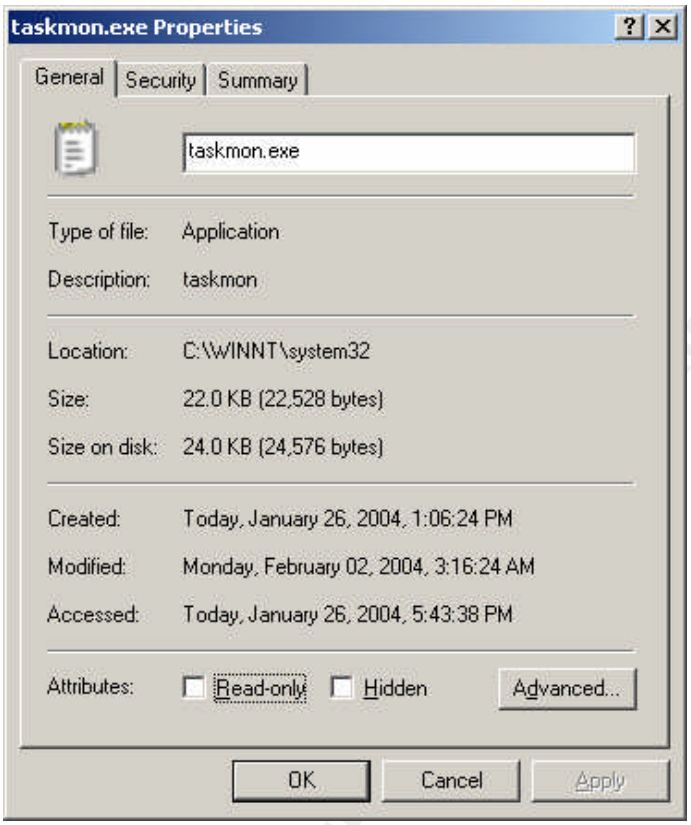

Figure 19 – taskmon.exe found in %System%

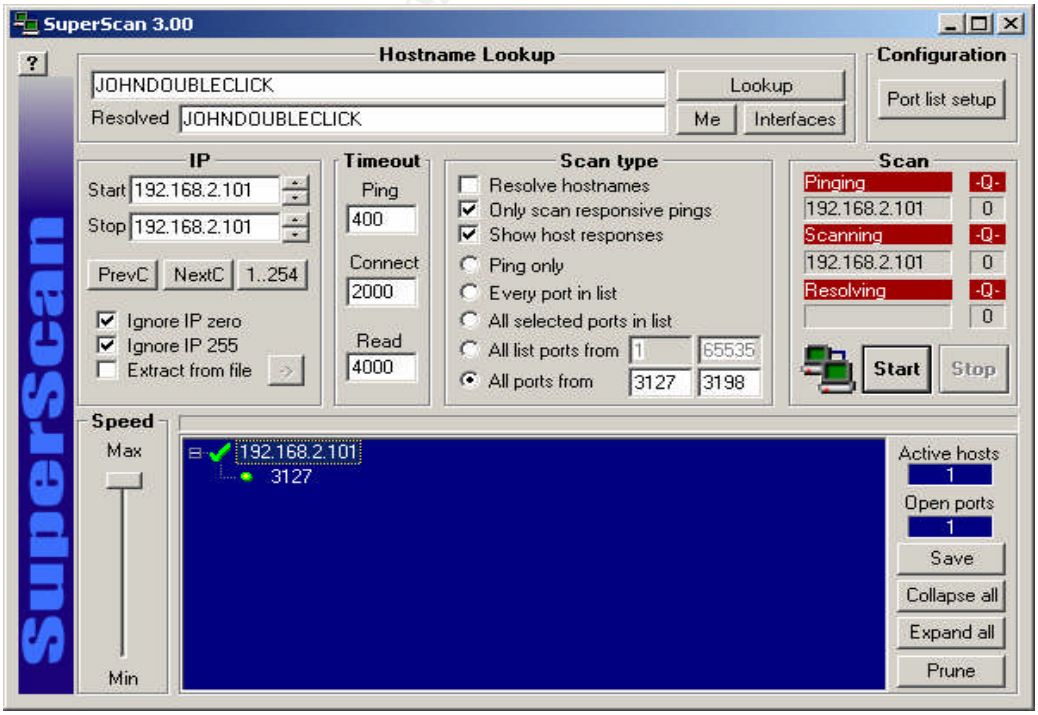

Figure 20 – Port 3127 open and active on John's PC

Kristof reported his findings to Mary and the rest of the Incident Handling team. This was now officially an incident and Mary established a CSIRT. Definition files were still in beta.

Timeline of events (Eastern Standard Time):

Monday January 26

8:36 AM
- Kristof discovers that a new virus called Mydoom is spreading in the wild (category 4).

10:46 AM
- Kristof continues to monitor activity and learns that various anti-virus vendors and news stations report that the global spread is rapid
- He is also aware that AV signature files have not yet been developed to mitigate the risk
- The rest of the Incident Handling team is made aware of Mydoom

12:48 PM
- Suspicious email sent from the Finances Department to Everybody at Viruses Unlimited with an attachment

1:02 PM
- Kristof questions the email sent at 12:48pm and advises with the Incident Handling team

1:06 PM
- John Doubleclick, CEO opens the attachment sent in the email

1:09 PM
- John calls his CFO to get more detail on the email sent out by the Finances Department

1:28 PM
- Mary calls the Manager of Finance to determine the validity of the email

1:33 PM
- Network Communications is alerted of increased traffic on the network

1:59 PM
- Mary has confirmation from the Manager of Finance that the email was not sent from anyone in the Finances Department

2:01 PM

- Mary sends an email to the Everybody distribution list advising them no to open the attachment in the email because of potential security risks

2:04 PM

- John calls Mary and tells her he opened the attachment

2:06 PM

- Kristof visits John's PC to scan for viruses and check patch compliance

2:07 PM

- Kristof learns that a pop-up text window was displayed when the attachment was opened….he suspects Mydoom

2:08 PM

- Kristof checks John's PC for Mydoom signatures

2:11 PM

- Kristof discovers that the files shimgapi.dll and taskmon.exe in %System% are present and that port 3127 is active and listening suggesting that a backdoor was successfully installed by Mydoom

2:13 PM

- Kristof reports his findings to the Incident Handling team

2:15 PM

- Mary Fary officially calls a CSIRT and the Incident Handling process is followed

3:13 PM

- Symantec releases tested definition files

3:42 PM

- Symantec releases Mydoom fix tool

5.3 Containment

When Mary got a phone call from John at 2:04pm regarding the email, the first thing she had him do was disconnect the network cable from his PC. That is when Kristof went to investigate his computer for anything suspicious. Although John had opened the attachment 58 minutes earlier this was the first line of containing the problem from the source. At this point the virus had already spread but disconnecting the network cable would prevent further spread from John and would ensure that an attacker could not connect to the backdoor on port 3127.

The Incident Handling team's "Jump Kit" is composed of the following:

- Knoppix bootable forensics CD
- Windows 2003 Resource Kit
- Several 'homemade' CDs with different software including nessus, nmap, superscan, netcat, ethereal, TCP dump, windump, mpack, dd, ghost, TCT forensics software, SSH, perl and several custom perl scripts
- Windows 98 bootable floppy disk
- Trinux bootable floppy disk
- 10 formatted floppy disks
- 10 CDRW
- Patch cables
- USB cable
- 8 port hub
- Iomega 120 GB USB 2.0 external desktop hard drive
- 802.11 card
- Blackberry RIM with Internet and phone enabled
- MP3 recorder
- Computer tool set
- Flashlight
- Notebook, pens and pencils
- Bags and ties

A follow-up company wide email was sent by Mary stating that the network was infected with Mydoom. She reiterated the fact not to open the email from the Finances Department as well as any other suspicious emails that may have arrived after 1:06 PM (the time John executed the virus). This would prevent further spread. Kristof was also continually scanning the network with a custom perl script looking for systems with Mydoom's version of taskmon.exe that was placed in the registry (Appendix B). Any system flagged was disconnected from the network by desktop technicians in each regional office and the switch port was disabled by the network engineers.

Kristof used an Iomega 120 GB USB 2.0 external desktop hard drive to backup John's system [30]. It is very simple to use and also very quick. Once the USB cable is plugged into the USB slot at the back of the PC the next available drive letter is shown on the screen, which represents the external drive. All data can be copied to this drive. The neat thing about this product is it also comes with Norton Ghost 2003. Therefore, Kristof also took an image of John's entire system (registry, etc.) and copied that to the external drive as well.

5.4 Eradication

As registry scans for taskmon.exe finished and all infected systems were pulled off the network, the lengthy clean-up process began. From the time John opened the attachment at 1:06pm to the time Symantec had released a fully tested definition file at 3:13pm, 232 systems were known to be infected. However, this number would probably keep increasing as people in the global regional offices were making their way to work in their respective time zones. Mary instructed Kelly Grolsch to immediately start pushing out the updated definition file to all servers and clients from the managed NAV server. This process would take about two hours. Once completed Kristof ran a perl script to scan for definition files older than January 26 (Appendix C). Any system flagged required a visit from a desktop technician for the updated signature file to be manually installed and the system fully scanned. First the system had to be disconnected from the network.

For systems that were already infected and off the network the Mydoom fix tool was run by a desktop support person to clean the system. It was a simple as saving the fix tool on a floppy and double-clicking the FxMydoom.exe file to start it [31] (Figures 21 and 22).

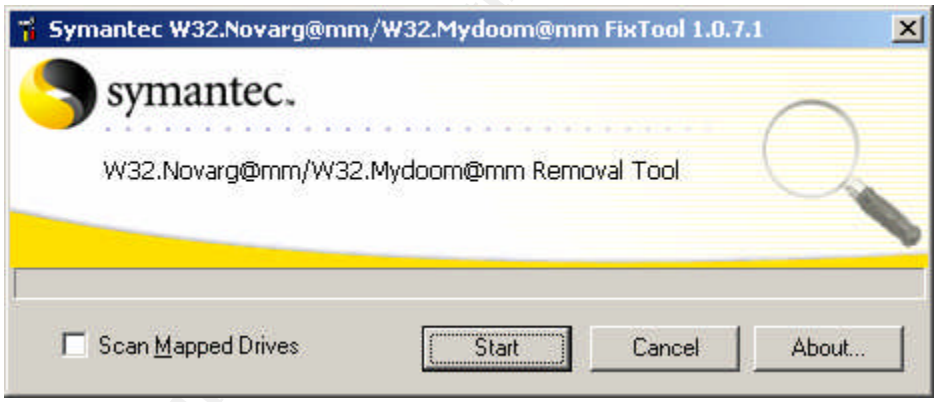

Figure 21 – Starting FxMydoom.exe

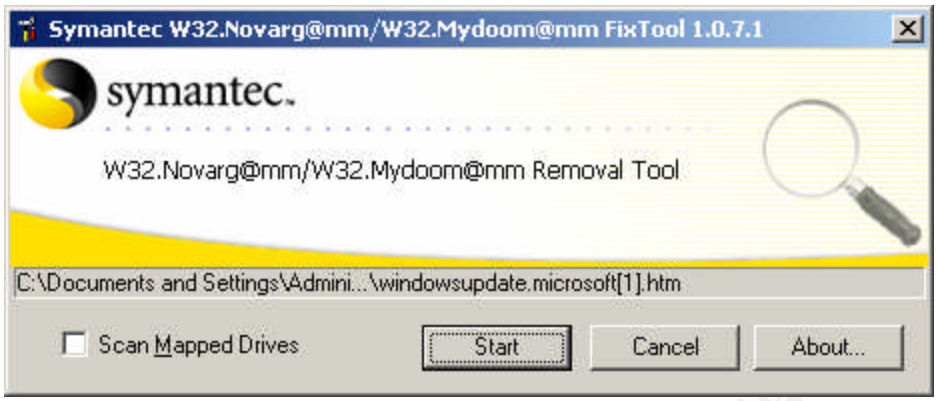

Figure 22 – Mydoom Fix tool cleaning the infection

Once the fix tool has completed its scan a window appears with summary statistics on what was detected, removed and fixed on the infected system (Figure 23).

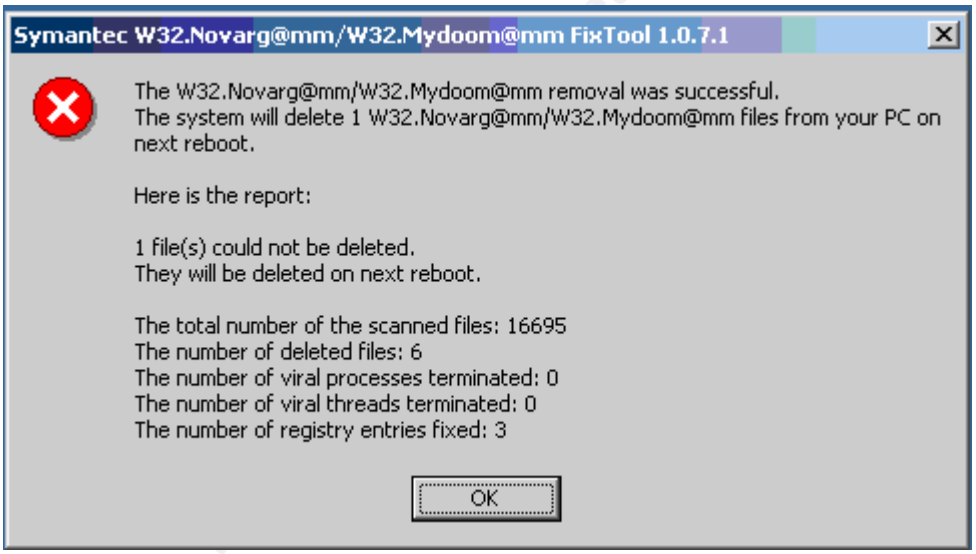

Figure 23 – Mydoom fix summary report

A more detailed text report is also provided with the filenames that were deleted and the registry entries that were fixed (Figure 24).

Figure 24 – Mydoom fix detailed report

Scanning for taskmon.exe and old definition files by Kristof and Brent would continue for the next several days until all infected systems were contained and cleaned. Scan times were setup as follows (EST):

- North America East Coast 10am – 1pm
- North America West Coast 1pm – 4pm
- UK 12pm – 1pm
- Asia/Australia 8pm – 11pm

The root cause of the spread of this virus was an uneducated user who did not think and opened a malicious attachment from a spoofed address. However, the true root cause is the person who created the spoofed message and specifically targeted the Viruses Unlimited network. Brent worked on trying to figure out who the attacker was and where this person launched the attack from. Was it an internal threat or external? He examined the image Kristof had created of John's system and initially did not see any traces of the attacker. He then proceeded to open his inbox and examine the spoofed message. Brent hoped the Internet Headers might give him a starting point. Sure enough, he was right. The Internet Headers provided some key information for the investigation (Figure 25).

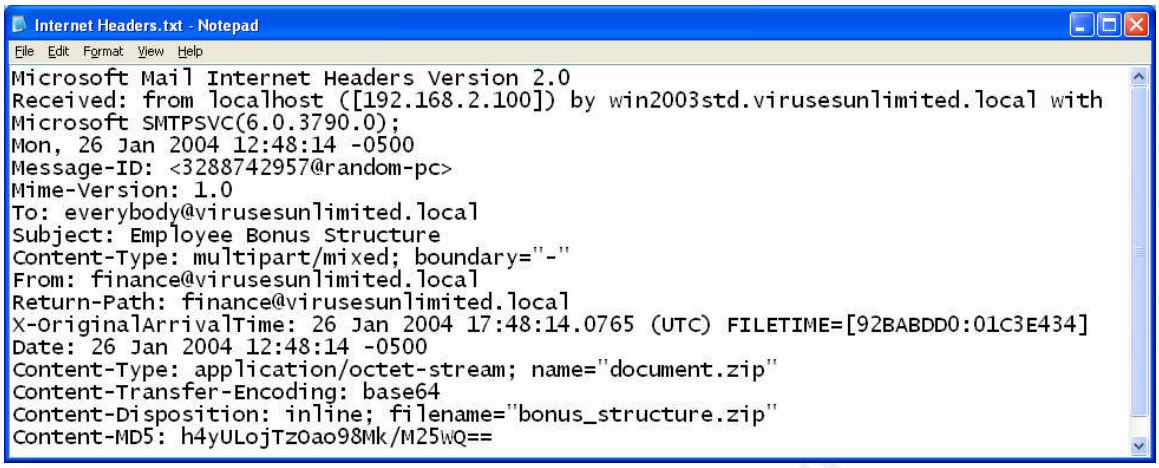

Figure 25 – Internet Headers from the spoofed email

The major observation Brent noticed was that there was an IP address associated with the sender of the email (192.168.2.100). The only hostname associated was localhost. However, the IP was enough to tell him that the attack was an internal one. He also noticed that the real filename of 'bonus_structure.zip' was 'document.zip'. This is one of the filenames that Mydoom is characteristic of.

Brent tried pinging the IP and running nbtstat –a to see if a name could be resolved as well as check if the host was alive on the network. Unfortunately, he ran into a brick wall. The next step was to work with the Network Engineers and see if they could possibly track down the last time this IP was on the network and the data port it was connected to. This was a long shot but worth a try.

It was determined that if the router cache and ARP tables had not timed out yet from the time this system disconnected from the network then this would be possible. It was also determined that switch logs could be reviewed to see which hosts joined and left the switch and correlate that back to the time when the virus was sent. The Network Engineers collaborated in a team effort and started searching through their logs. Sure enough the IP 192.168.2.100 was discovered as joining the network at 11:48am and disconnecting at 1:34pm. The switch port was traced back to a data port in the executive meeting room. This information was relayed to the Incident Handlers in a timely manner. They found it a little odd that the email was sent from the executive meeting room since only executives were allowed to book meetings there and none were scheduled between the times stated above. Mary sent an email to senior management asking if any of them were in the meeting room from 11:30am to 1:30pm or if they had noticed anyone other than senior management in there. All replies were negative. She then sent out a similar email to all of the people in the office space where the incident occurred. After about 10 minutes the receptionist emailed back stating that she had escorted a gentleman into the room around 11:40am who was to meet with the CEO. Things started making a bit more

sense to the Handlers. Mary called the receptionist and asked why the man was allowed in without proper supervision. She had no response. Next she asked the receptionist if she had any information on whom this individual was. The receptionist scrambled around for a couple minutes as she tried to find the card where she scribbled down the man's name. His name was Joe Infect and that is all she had. Brent, Kelly and Kristof looked up Joe on the Internet and discovered that he was working for a company who was in direct competition with Viruses Unlimited. This same company recently lost a multi million-dollar sale to Viruses Unlimited. The Incident Handling team gathered as much information as possible on Joe and contacted local law enforcement to follow-up with the matter. Viruses Unlimited lawyers offered to cooperate with local authorities and suggested that any hardware owned by Joe Infect be confiscated for forensics purposes.

5.5 Recovery

Once the fix tool has been run and the system is clean of Mydoom the current definition file from January 26th is installed and a full system scan is performed (Figure 26). Kristof ran this scan manually but did schedule a daily scan on John's PC to run each day at 5pm. He also checked the NAV parent server to make sure that it was polling John's system every hour for the most current definition file. NAV 8.x does this by default.

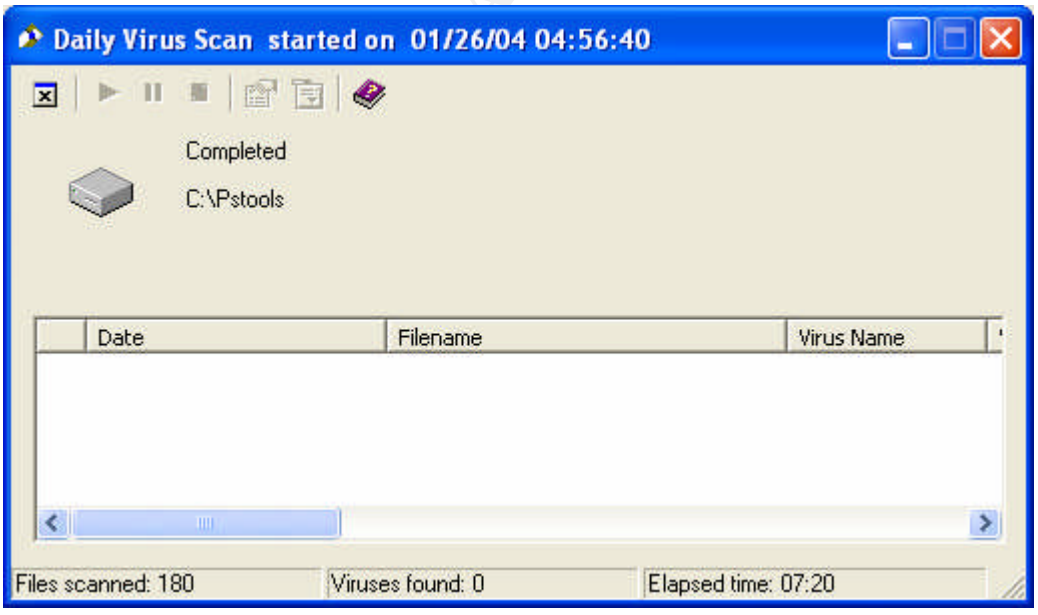

Figure 26 – Completed virus scan showing zero viruses

As a double check the Mydoom fix tool is run again to ensure John is not infected (Figure 27). At this point the system is in a "known good" state and placed back on the network.

Figure 27 – Mydoom fix tool states no virus found

Users that were not in the office during the day of the outbreak would have had their signature files updated when they returned the next day. When Project Manager, John Smith checked his email upon returning on January 27th he saw the spoofed email from the Finances Department as well as a random email sent by an infected system (Figure 28). Of course this was one way that Mydoom propagated.

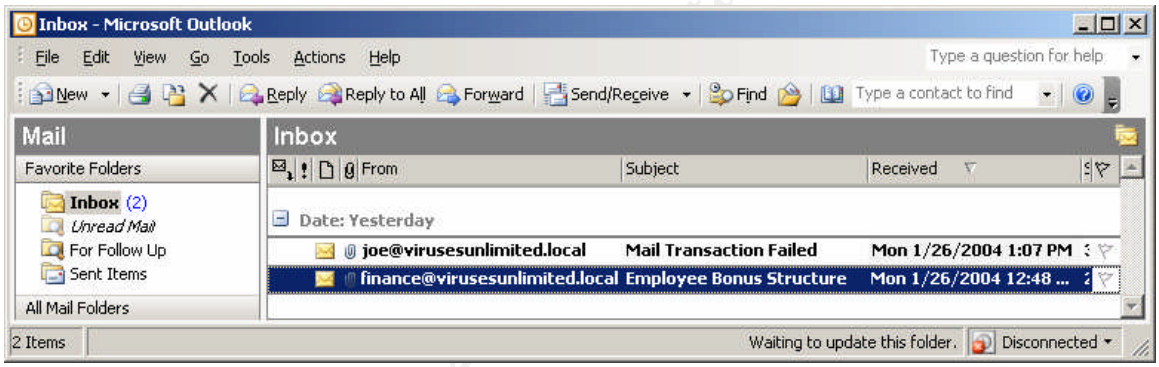

Figure 28 – Spoofed email sent by an infected system

Figure 29 shows the message body and the attached virus that was sent by the infected system. Outlook 2003 blocks unsafe attachments by default, which is a handy feature.

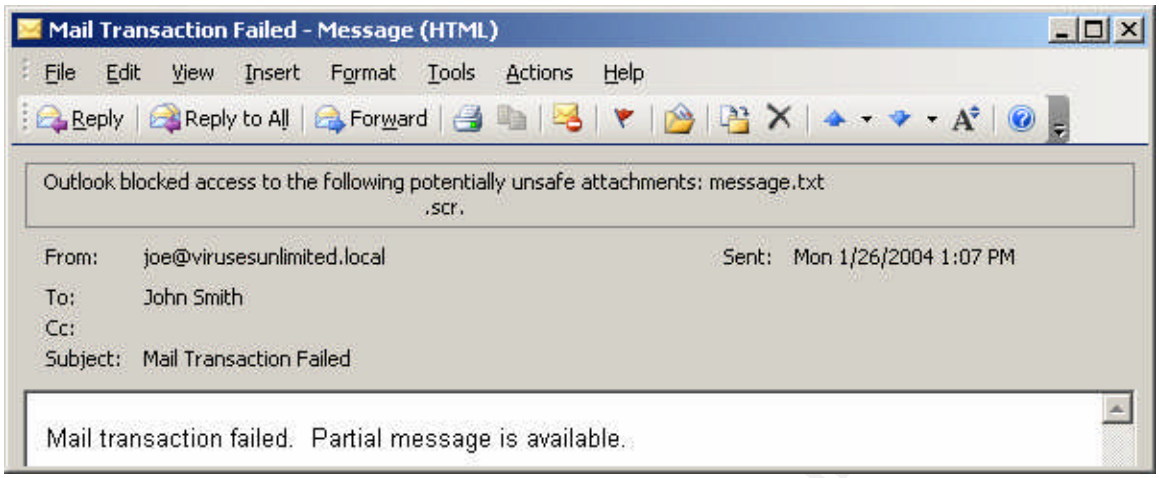

Figure 29 – Blocked unsafe attachment sent by an infected system

John Smith had not yet read Mary's emails regarding the spoofed email from the Finances Department. Like many of his colleagues, temptation got the best of him and he opened the attachment that came with the message. However, because his signature files were up to date Mydoom was intercepted and quarantined (Figure 30).

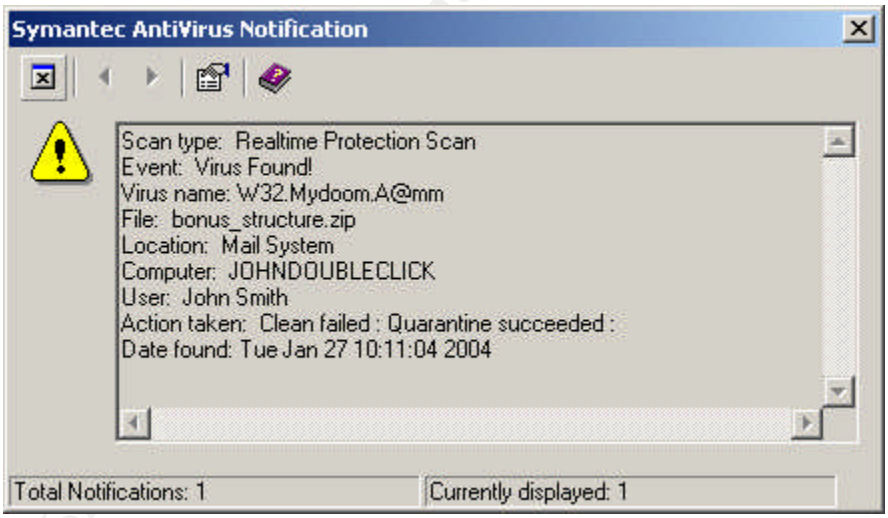

Figure 30 – Mydoom intercepted and quarantined by Symantec Antivirus

Systems with updated signature files will be protected against the latest virus/worm threats as well as all past ones. Infected systems are pulled off the network, cleaned with the fix tool, have the current signature file installed at which point a full system scan is run and finally the fix tool is run again. Once confirmed clean the system is placed back on the network.

5.6 Lessons Learned

Viruses Unlimited had done well in terms of preparing themselves for their next incident because of lessons learned from past incidents. Their network was locked down from the perimeter, policies were in place and the Incident Handling process was smooth and consistent. This particular incident was handled according to process and the problem contained and eradicated in just less than 48 hours time. However, the million dollar question at the post incident meeting was 'why did we (Viruses Unlimited) get infected this time if policies and processes are in place'? The answer; it is quite obvious that the possibility of an internal threat was ignored. Had policies 17 through 26 (outlined in the Preparation phase) been followed this incident would have been avoided. How can this be improved upon for next time a 'stranger' enters the office building? Below are the key points taken from the meeting and post incident report to prevent this from happening again.

User Education and Awareness of Policies, Procedures and Enforcement

- Setup training programs within the organization to teach all users (including all levels of management) on existing policies, why they exist and their importance. This must be ongoing.
- Educate all users on the business risks of breaching a policy and the value of assets, corporate information and the potential damage to the organization if this information becomes public. It is crucial to have buy-in from Senior Executives to fund this type of training and mandate it.
- Publish policies on the company intranet and budget for software that forces a user to be quizzed on company policies. This ensures they have read and understood the policy. Also, an electronic user signature is stored in a database if required for future legal action due to an internal breach.

Regular Auditing on the Enforcement Operations

- Physical security is often ignored or not deemed important within an organization. Perform physical security audits once a month.
- Attempt to enter different office locations within the corporation without a security pass card. Use social engineering techniques and masquerade yourself. Try being a 'different' person to see who succeeds and who fails. For example, try different outfits and personalities; a well groomed business person, a courier person, a handy-man coming to fix the coffee machine.
- Attempt to plug into the corporate network with a personal laptop and browse the network.
- Attempt to walk out of the office with a piece of hardware.
- Test out help desk staff to see how easy it is to get domain passwords or voice mailbox passwords.

- Walk around and do an inventory on the number of workstations that are unlocked. Sit down at the desk to see what kind of valuable information you can dig up. Read a few emails and forward them to your account but delete the email from the 'sent items' and the 'deleted items' folders.
- Budget for video cameras in strategic locations.
- Periodically send out suspicious looking emails to groups of users and monitor who opens them.

Note: Sign-off was required by management at Viruses Unlimited to perform the above security audits.

In summary, Viruses Unlimited was well prepared for their next incident. Policies were in place, the Incident Handling process was solid and from the perimeter they were locked down tightly. Where they failed was user awareness on existing policies and lack of physical security in their environment. Had policies been followed and physical security been enforced and audited then the spread of Mydoom would not have happened.

6.0 REFERENCES

[1] Ferrie, Peter, and Tony Lee. "Symantec Security Response, W32.Mydoom.A@mm". 26 February, 2004. URL: http://securityresponse.symantec.com/avcenter/venc/data/w32.novarg.a@mm.html (26 Jan. 2004).

[2] Carnegie Mellon University. "CERT® Incident Note IN-2004-01". 27 January, 2004. URL: http://www.cert.org/incident_notes/IN-2004-01.html.

[3] Internet Assigned Numbers Authority. "Port Numbers". 28 April, 2004. URL: http://www.iana.org/assignments/port-numbers.

[4] Klensin, J. "RFC 2821". April, 2001. URL: http://rfc.net/rfc2821.html.

[5] Mitchell, Bradley. "P2P - Peer to Peer Guide picks". 16 March, 2004. URL: http://compnetworking.about.com/cs/peertopeer/.

[6] Lowth, Chris. "Securing Your Network Against Kazaa". 1 September, 2003. URL: http://www.linuxjournal.com/article.php?sid=6945.

[7] Bandwidth Controller, Bandwidth Management Software. "Limit Kazaa Bandwidth Usage". Date not listed. URL: http://bandwidthcontroller.com/limit-kazaa-bandwidth.html.

[8] Information Sciences Institute. "RFC 793". September, 1981. URL: http://www.freesoft.org/CIE/RFC/793/.

[9] Gudmundsson, O. "RFC 3658". December, 2003. URL: http://www.armware.dk/RFC/rfc/rfc3658.html.

[10] Fielding, et al. "RFC 2616". June, 1999. URL: http://www.w3.org/Protocols/rfc2616/rfc2616.html.

[11] Symantec Security Response. "Symantec Security Response Threat Writeups (2026)". Dates vary according to virus release date. URL: http://securityresponse.symantec.com/avcenter/venc/auto/index/indexW.html.

[12] 6.824 Distributed Computer Systems. "6.824 Lab 3: A TCP proxy". Fall, 2004. URL: http://www.pdos.lcs.mit.edu/6.824/labs/tcpproxy.html.

[13] McAfee Security. "W32/Mydoom@MM". 11 March, 2004. URL: http://vil.nai.com/vil/content/v_100983.htm (26 Jan. 2004).

[14] Matt D., J. Todd. and S. Sonu. "See Everything with Ethereal!". 1 May, 2003. URL: http://www.diversedev.com/misc/bcomm/manual.doc.

[15] Computer Associates, Virus Information Center. "Win32.Mydoom.A". 12 April, 2004. URL: http://www3.ca.com/threatinfo/virusinfo/virus.aspx?id=38102 (26 Jan. 2004).

[16] Carrera, Ero, and Gergely Erdelyi. "F-Secure Virus Descriptions : Mydoom". 28 January, 2004. URL: http://www.f-secure.com/v-descs/novarg.shtml.

[17] Sophos. "W32/MyDoom-A". 26 January, 2004. URL: http://www.sophos.com/virusinfo/analyses/w32mydooma.html.

[18] Kaspersky. "I-Worm.Mydoom (also known as Novarg)". 26 January, 2004. URL: http://www.avp.ch/avpve/worms/email/novarg.stm.

[19] de Mata, Aldrich. "WORM_MYDOOM.A". 28 January, 2004. URL: http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_MYDOOM.A&VSect=T (26 Jan. 2004).

[20] Sam Spade.org. "Sam Spade.org". URL: http://www.samspade.org/.

[21] ARIN. "American Registry for Internet Numbers". URL: http://www.arin.net.

[22] Netcraft. URL: http://news.netcraft.com/.

[23] Foundstone. "Free Tools". Version 4.0. URL: http://www.foundstone.com/index.htm?subnav=resources/navigation.htm&subcontent=/resources/proddesc/superscan.htm.

[24] Insecure.org. "Featured News, Nmap 3.50 Released". Version 3.5.0. URL: http://www.insecure.org/.

[25] http://linux.maruhn.com. "Mpack - Mpack and munpack MIME e-mail utilities". 15 September, 2003. URL: http://linux.maruhn.com/sec/mpack.html.

[26] Giacobbi, Giovanni. "The GNU Netcat project". Version 0.7.1. 27 February, 2004. URL: http://netcat.sourceforge.net/.

[27] http://windump.polito.it/. "WinDump: tcpdump for Windows". Version 3.6.2. 2 January, 2004. URL: http://windump.polito.it/.

[28] http://windump.polito.it/. "WinDump Manual". 14 March, 2002. URL: http://windump.polito.it/docs/manual.htm.

[29] SANS and Ed Skoudis. "Incident Handling Step-by-Step and Computer Crime Investigation". SANS Institute Track 4 – Hacker Techniques, Exploits & Incident Handling. February 2004 (2003): 35 – 111.

[30] http://www.iomega.com/. "Iomega HDD 120GB USB 2.0/Firewire External Desktop Hard Drive". URL: http://www.iomega.com/na/products/product_detail.jsp?PRODUCT%3C%3Eprd_id=5274255&FOLDER%3C%3Efolder_id=63237&ASSORTMENT%3C%3East_id=63191&bmUID=1084909353372.

[31] http://www.symantec.com/. "W32.Mydoom@mm Removal Tool". 5 March, 2004. URL: http://securityresponse.symantec.com/avcenter/venc/data/w32.mydoom@mm.removal.tool.html (26 Jan. 2004).

APPENDIX A – Mydoom Characteristics

Mydoom searches for looks for files with the following extensions:

- asp
- dbx
- tbb
- htm
- sht
- php
- adb
- pl
- wab
- txt

Message subject may be one of the following:

- test
- hi
- hello
- Mail Delivery System
- Mail Transaction Failed
- Server Report
- Status
- Error

Message body may be one of the following:

- test
- The message cannot be represented in 7-bit ASCII encoding and has been sent as a binary attachment.
- The message contains Unicode characters and has been sent as a binary attachment.
- Mail transaction failed. Partial message is available.

Attachment name may be one of the following:

- document
- readme
- doc
- text
- file
- data
- test
- message
- body

Attachment extension may be one of the following:

- pif
- scr
- exe
- cmd
- bat
- zip

Note: Sender's address is random and probably spoofed

APPENDIX B – PERL Script - scan for taskmon.exe

```
#! d:\perl\bin\perl.exe
# mydoom_scan.pl
# Combs through Registry keys, looking for taskmon.exe in %System%
# usage: perl mydoom.pl [network segment]

use strict;
use Net::hostent;
use Win32::TieRegistry( Delimiter=>"/", ArrayValues=>0 );
use Net::Ping;

# collect up all arguments to the script and parse through them one at a time
        foreach my $arg (@ARGV) {

# append a trailing dot unless there's already one there
        $arg = $arg."\." unless ($arg =~ m/\.$/);

# RegEx for class C IP range
        if ($arg =~ m/^\d{1,3}\.\d{1,3}\.\d{1,3}\.$/) {
                foreach my $i (2..254) {
                my $ip = $arg.$i;

# Now that we have our IP address, this is where we do stuff
                my $p = Net::Ping->new("icmp");
                        if ($p->ping($ip,2)) {
                        \&checkKeys($ip);
                        }
#print "\n";
                }
        }
}


sub checkKeys {

# Default to local machine if no machine name given
                my $server = $_[0];
                print "\n$server~\   ";

# Get hostname
                if (my $h = gethost($server)) {
                my $name = $h->name;
                print "$name~\   ";
                }
```

```
#               else {
#               print "Could not connect to hive";
#               }

# Registry key to check
                my $nw = "SOFTWARE/Microsoft/Windows/CurrentVersion/Run/";

# Connect to (remote) Registry
                if (my $remote = $Registry->{"//$server/LMachine"}) {

# Connect to Registry key
                if (my $data = $remote->{$nw}) {

# Enumerate all subkeys
                my @subkeynames = $data->SubKeyNames();

# Enumerate through all subkeys
                foreach my $subkey (@subkeynames) {
                        if (my $str = $data->{"$subkey/TaskMon"}) {

# Search for strings in the CurrentVersion TaskMon string
                if (grep(/(taskmon.exe)/i,$str)) {
                print "$str~ MYDOOM FOUND!!!!!\ ";
                }

                else {
                print "MYDOOM NOT FOUND!!!!~";
                }
                }

                else {
                print "Could not connect to $subkey~";
                }
        }
}

                else {
                print "Could not connect to key $nw~";
        }
}

                else {
                print "Could not connect~";
                }
        }
```

APPENDIX C – PERL Script - scan for old definition files

```
#! d:\perl\bin\perl.exe
# defs.pl
# Combs through Registry keys, looking for out of date NAV definiton files
# usage: perl defs.pl [network segment]

use strict;
use Net::hostent;
use Win32::TieRegistry( Delimiter=>"/", ArrayValues=>0 );
use Net::Ping;

# collect up all arguments to the script and parse through them one at a time
        foreach my $arg (@ARGV) {

# append a trailing dot unless there's already one there
        $arg = $arg."\." unless ($arg =~ m/\.$/);

# RegEx for class C IP range
        if ($arg =~ m/^\d{1,3}\.\d{1,3}\.\d{1,3}\.$/) {
                foreach my $i (2..254) {
                my $ip = $arg.$i;

# Now that we have our IP address, this is where we do stuff
                my $p = Net::Ping->new("icmp");
                        if ($p->ping($ip,2)) {
                        \&checkKeys($ip);
                        }
#print "\n";
                }
        }
}


sub checkKeys {

# Default to local machine if no machine name given
                my $server = $_[0];
                print "\n$server`\   ";

# Get hostname
                if (my $h = gethost($server)) {
        my $name = $h->name;
                print "$name`\   ";
                }
```

```
# Registry key to check
	my $nw = "SOFTWARE/Symantec/";

# Connect to (remote) Registry
	if (my $remote = $Registry->{"//$server/LMachine"}) {

# Connect to Registry key
	if (my $data = $remote->{$nw}) {

# Enumerate all subkeys
	my @subkeynames = $data->SubKeyNames();

# Enumerate through all subkeys
	foreach my $subkey (@subkeynames) {
	if (my $str = $data->{"$subkey/DEFWATCH_10"}) {

# Search for strings in the SharedDefs Value
	if (grep(/(20040126)/i,$str)) {
	print "DEF FILE UPDATED--$str\` ";
	}
		else {
		print "DEF FILE OUT OF DATE--$str\` ";
	}
}
#		else {
#		print "Could not connect to $subkey`";
#		}
	}
}
		else {
		print "Could not connect to key $nw`";
	}

}
		else {
		print "Could not connect to hive`";
	}
}
```