Bugbear takes a bite out of Novell – twice

By Christopher Rowe

GCIH Practical Assignment Version 2.1a, Option 1
June 5, 2003

INTRODUCTION

Fool me once, shame on you. Fool me twice, shame on me. Yet, twice in a span of two weeks, an educational organization, herein referred to as College, was hit twice by the same worm, the infamous W32.Bugbear@mm, or, simply, Bugbear. A curious report of printers spewing out paper overnight led to the internal discovery of one of the most advanced and fast-spreading worms to date. To date, there have been nearly 700,000 reports of infections, with 100,000 of those infections coming in the first 24 hours of discovery, and 200,000 within the first four days.[1] Although, most antivirus vendors have downgraded the threat from this worm, the 500,000 additional cases reported over the last eight months makes it clear that this worm is still a threat. Additionally, one particular aspect of this worm's behavior, planting a backdoor Trojan, may still be prevalent in many systems. Thus the threat has not been completely eradicated.

This paper seeks to explain the Bugbear worm incidents at this organization, and provide a guide for better incident handling in the future. This project serves as the practical for the GIAC Certification for Incident Handling, and is written according to GCIH Practical Guidelines v. 2.1.

PART ONE
The Exploit

NAME: All major antivirus software providers, including Symantec, Trend Micro, Command, et al, have identified the Bugbear worm. For a list of individual vendors various names, see the chart below. The vulnerability that Bugbear is based on is identified in CVE-2001-0154 and Microsoft Security Bulletin MS01-020. These are explained below.

Antivirus software identifications by vendor

VENDOR	VIRUS NAME
Symantec	W32.Bugbear@mm
Trend Micro	Worm_Bugbear.A
F-secure	Tanatos
Computer Associates	Win32/Bugbear.worm
McAfee	W32/Bugbear@MM

[1] http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_BUGBEAR.A&VSect=S&Period=All

OPERATING SYSTEMS AFFECTED: Research indicates this worm targets most Microsoft Windows operating systems. However, It should be noted that, as the title indicates, this worm affected a Novell 4.11 server, running Groupwise 5.5. Despite seemingly being inoculated from the worm because of the choice NOS and e-mail client software, Groupwise is just as susceptible to it for the same reasons as Outlook and Outlook Express. According to one Groupwise reference, Administering Groupwise 5.5, "Groupwise now supports some HTML editing within message view … [it] has dependency on Microsoft 4.x or better."[2] In short, Groupwise's dependency on IE for HTML renderings opens it up to the same vulnerabilities as Microsoft clients. Additionally, once a machine is infected with the worm, it can continue to propagate through file shares.

While most research on the Bugbear worm indicates that the exploit affects Outlook and Outlook Express, no workstations in the College network used either program as its e-mail client. Instead, all used the Groupwise client software through ZenWorks, running on top of various Windows desktops (Windows 95, Windows 98SE, Windows 2000, Windows XP). These systems range anywhere from Compaqs with PII processors (266 mhz) and 32 MB of RAM, running Windows 95, to brand new Dell Optiplex G260s (P4, 1.8 Ghz, 512 MB), and everything in between (see **PART TWO: The Attack: DESCRIPTION AND DIAGRAM OF THE NETWORK:** for specific information on affected systems).

Curiously, Symantec completely ignores the Novell Netware operating system in listing vulnerable and non-vulnerable OSs. While the site specifically claims Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Me may be affected, and plainly states that Macintosh, Unix and Linux are NOT affected, Netware is not mentioned. However, the following comment is found in the instructions for removal of the virus:

> This tool is not designed to run on Novell NetWare servers. To remove this threat from a NetWare server, after making sure that you have current virus definitions, run a full system scan with your Symantec antivirus product.

Note, also, that the site claims Linux is not affected, although many reports have stated that Linux print servers have been subjected to the same "garbage printing" by this virus as Windows servers.[3]

A patch for Internet Explorer that fixes this vulnerability is available from Microsoft and is included individually or as part of IE Service Pack 2. There are

[2] Administering Groupwise 5.5. Tayler, Kratzer, Phillips. McGraw-Hill. 0-07-212329-x

[3] http://securityresponse.symantec.com/avcenter/venc/data/w32.bugbear@mm.html

numerous fixes for Microsoft Outlook as well, and MS Outlook SP2 and MS Outlook XP SP1 will block the attachment download as well.[4] Click here for the patches.

PROTOCOLS: Bugbear uses and exploits several protocols, including MIME, NetBIOS, SMB, SMTP, LPR, along with HTTP/FTP, TCP/IP, IMAP/POP3 and API. The worm takes advantage of a vulnerability in Internet Explorer, versions 5.01 and 5.5. According to CVE-2001-0154:

> [An] HTML e-mail feature in Internet Explorer 5.5 and earlier allows attackers to execute attachments by setting an unusual MIME type for the attachment, which Internet Explorer does not process correctly.[5]

Microsoft also has documented this vulnerability in Microsoft Security Bulletin MS01-020, originally posted March 28, 2001.

> Because HTML e-mails are simply web pages, IE can render them and open binary attachments in a way that is appropriate to their MIME types. However, a flaw exists in the type of processing that is specified for certain unusual MIME types. If an attacker created an HTML e-mail containing an executable attachment then modified the MIME header information to specify that the attachment was one of the unusual MIME types that IE handles incorrectly, IE would launch the attachment automatically when it rendered the e-mail.[6]

Interestingly, Scott Winters discussed this topic in-depth two years ago in a GCIH practical paper. However, as Winters noted, there were not any actual exploits at the time, only potential for them:

> Despite the fact that no known malicious code has been released that takes advantage of this vulnerability, due to its incredible potential, it is imperative for continued system security that this patch is downloaded and applied as soon as possible. [7]

Well, the Black Hats have never been ones to let us down, and today, there are numerous exploits, including the Bugbear virus discussed herein.

BRIEF DESCRIPTION: The Bugbear worm attacks in a variety of ways. It is a C++ program that has its own SMTP engines built in, thus using e-mail as one

[4] http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/virus/alerts/bugbear.asp

[5] http://www.cve.mitre.org/cgi-bin/cvekey.cgi?keyword=cve-2001-0154

[6] http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS01-020.asp

[7] http://www.giac.org/practical/Scott_Winters_GCIH.doc

means of propagating. The use of SMB/NetBios allows it to replicate through network shares and spread throughout the network. Once it finds a machine to victimize, it copies itself into the system, giving itself a random name, adding a startup key to the Registry. In addition, a keylogger is dropped into the System folder. This allows the attacker to record the keystrokes on the infected machines, and retrieve this information through a backdoor Trojan that is planted by the worm and opened through TCP port 36794. The program also attempts to terminate various processes, including several antivirus programs. Finally it creates three encrypted DLL files and two DAT files.[8]

VARIANTS: Just before submission of this paper, a new variant has emerged. W32/Bugbear.B@mm (F-Secure) apparently has similar attributes to the original Bugbear worm. However, this variant is polymorphic, making this new worm even more difficult to track and contain.

In addition, the Jdbgmgr.exe File Hoax takes advantage of the Bugbear outbreak to trick users into deleting the Windows application Jdbgmgr.exe (%systemroot%\i386\jdbgmgr.exe), which has a teddy bear icon (see right). This file is NOT a virus: it is a legitimate, harmless file for debugging Java applets that works with Internet Explorer, thus deletion of Jdbgmgr.exe can hinder the execution of certain Java applets.[9] (Furthermore, W32.Efortune.31384@mm actually targets the Jdbmgr.exe file.[10]) Though hoaxes themselves are not problematic to the system, the social engineering that they can trigger, such as this, are just as dangerous, arguably more so, since they can be more difficult to recover from.

REFERENCES*:*

http://www.f-secure.com/bugbear/

ftp://ftp.rfc-editor.org/in-notes/rfc821.txt

ftp://ftp.rfc-editor.org/in-notes/rfc822.txt

ftp://ftp.rfc-editor.org/in-notes/rfc1652.txt

[8] http://www.europe.f-secure.com/v-descs/tanatos.shtml

[9] http://securityresponse1.symantec.com/sarc/sarc-intl.nsf/html/br-jdbgmgr.exe.file.hoax.html

[10] http://securityresponse.symantec.com/avcenter/venc/data/w32.efortune.31384@mm.html

http://securityresponse.symantec.com/avcenter/venc/data/w32.bugbear@mm.html

http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_BUGBEAR.A

http://www.cve.mitre.org/cgi-bin/cvekey.cgi?keyword=cve-2001-0154

http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS01-020.asp

http://securityresponse1.symantec.com/sarc/sarc-intl.nsf/html/br-jdbgmgr.exe.file.hoax.html

http://securityresponse.symantec.com/avcenter/venc/data/w32.efortune.31384@mm.html

http://www.zzee.com/email_security/

http://vil.nai.com/vil/content/v_99728.htm

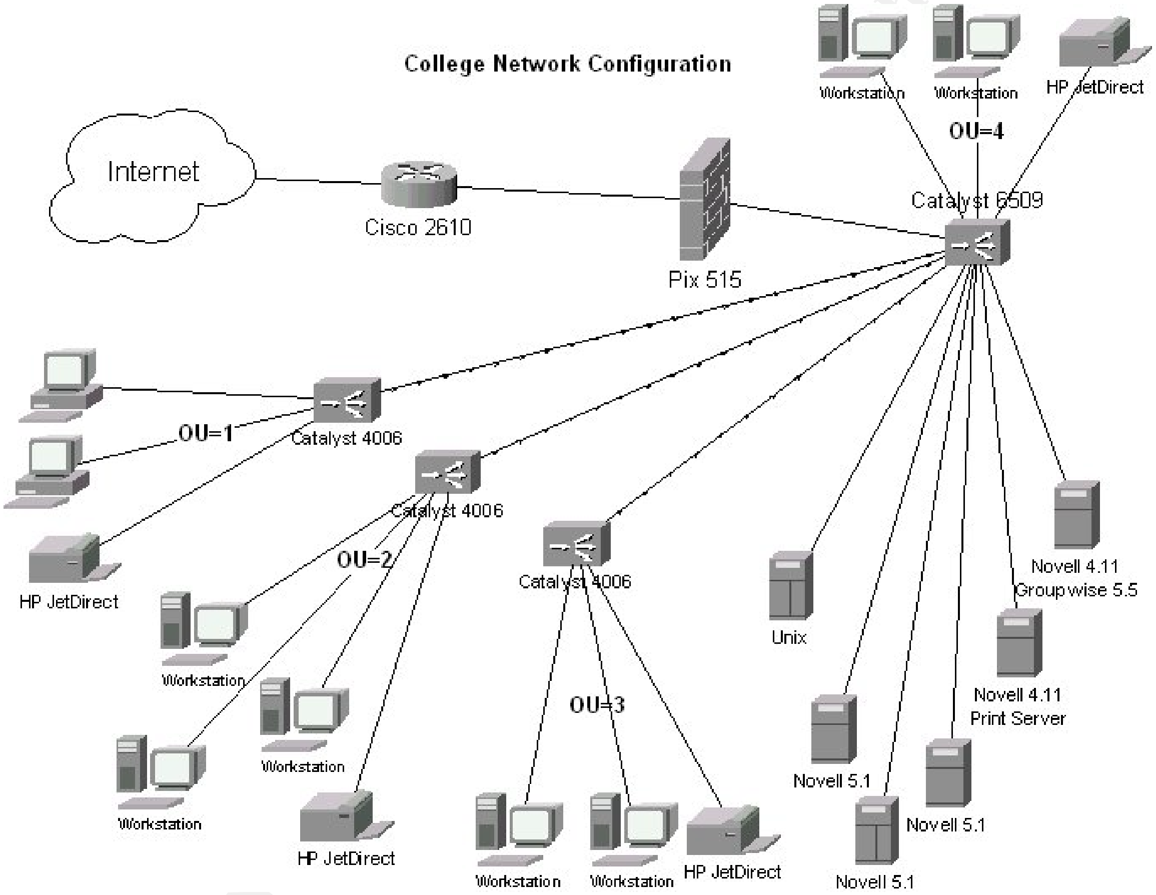
http://www.mhonarc.org/~ehood/MIME/toc.html

Andrews, Jean. i-NET+ Guide to the Internet, Second Edition. Course Technology, 2002. 0-619-12068-1. pps. 186-193.

Tayler, Howard, Ross Phillips and Tay Kratzer. Administering Groupwise 5.5. McGraw-Hill Osborne Media; 1st edition, 2000. 0-07-212329-x

PART TWO
The Attack

DESCRIPTION AND DIAGRAM OF THE NETWORK: Despite the physical size of the network, spanning a half-mile campus, and the logistical size, with 100s of employee users and thousands of students, the network design is relatively straightforward. Essentially, it is a switched-star topology.



A single Cisco 2610 connects the network to the Internet. Traffic coming into the network is routed into the network, and then filtered through a Cisco Pix 515 firewall (See below for partial rule set). From there, acceptable traffic is forwarded to a Catalyst 6509 which uses a backplane routing module and hardware switching capabilities to forward traffic to the appropriate LAN segment. The Catalyst 6509 connects the server group through Ethernet, while other segments, located in other buildings on campus, are connected via fiber. Each building is divided into its own subnet.

The firewall is configured to block unnecessary traffic, including ICQ, AIM, Net Meeting, etc. from going out, and uses a “Deny any except what is specifically permitted” philosophy for filtering inbound traffic.

Partial Ruleset for Pix 515E-R:

```
conduit permit tcp host MAIL eq smtp any
conduit permit tcp host WEB eq www any
conduit permit tcp host MAIL eq www any
conduit permit tcp host MAIL eq 1677 any
conduit permit tcp host MAIL eq 1678 any
conduit permit tcp host MAIL eq 7100 any
conduit permit tcp host HOST1 eq www any
conduit permit tcp host HOST2 eq www any
conduit permit tcp host HOST2 eq www any
conduit permit tcp host MAIL eq 7205 any
conduit permit tcp host MAIL eq 7648 any
conduit permit tcp host MAIL eq 8008 any
conduit permit tcp host MAIL eq 8009 any
conduit permit tcp host MAIL eq 2200 any
conduit permit udp host MAIL eq 1677 any
conduit permit tcp host x.x.x.x any
conduit permit esp host ford any
conduit deny tcp host x.x.x.x any
conduit permit tcp kent x.x.x.x any
conduit permit tcp host x.x.x.x eq 1214 any
conduit permit udp host x.x.x.x eq 1214 any
conduit permit tcp any eq 1723 host x.x.x.x
conduit permit udp any eq 1723 host x.x.x.x
conduit permit gre any host x.x.x.x
conduit deny icmp any any
conduit permit udp any eq isakmp host x.x.x.x
conduit permit udp any eq 259 host x.x.x.x
conduit permit udp any eq 2746 host x.x.x.x
conduit permit tcp any eq 18207 host x.x.x.x
conduit permit tcp host cis eq 5500 any
conduit permit tcp host cis eq 5700 any
outbound  1 deny 0.0.0.0 0.0.0.0 5190-5193 tcp
outbound  1 deny 0.0.0.0 0.0.0.0 5210-5235 tcp
outbound  1 deny 0.0.0.0 0.0.0.0 4000 udp
outbound  1 deny 0.0.0.0 0.0.0.0 6970-7170 udp
outbound  1 deny 0.0.0.0 0.0.0.0 1007-1008 udp
outbound  1 deny 0.0.0.0 0.0.0.0 1007-1008 tcp
outbound  1 except 0.0.0.0 0.0.0.0 20 tcp
outbound  1 except 0.0.0.0 0.0.0.0 21 tcp
outbound  1 except 0.0.0.0 0.0.0.0 23 tcp
outbound  1 except 0.0.0.0 0.0.0.0 25 tcp
outbound  1 except 0.0.0.0 0.0.0.0 53 tcp
outbound  1 except 0.0.0.0 0.0.0.0 110 tcp
outbound  1 except 0.0.0.0 0.0.0.0 80 tcp
outbound  1 except 0.0.0.0 0.0.0.0 443 tcp
outbound  1 permit teribost 255.255.255.255 2032 tcp
outbound  1 permit teribost 255.255.255.255 992 tcp
outbound  1 permit teribost 255.255.255.255 993 tcp
outbound  1 permit teribost 255.255.255.255 8999 tcp
outbound  1 permit teribost 255.255.255.255 2032 udp
outbound  1 permit teribost 255.255.255.255 992 udp
outbound  1 permit teribost 255.255.255.255 993 udp
outbound  1 permit teribost 255.255.255.255 8999 udp
outbound  1 except 0.0.0.0 0.0.0.0 53 udp
outbound  1 permit friesc 0.0.0.0 1677 tcp
outbound  1 permit friesc 0.0.0.0 1677 udp
outbound  1 deny 0.0.0.0 0.0.0.0 1214 tcp
outbound  1 deny 0.0.0.0 0.0.0.0 6346 tcp
outbound  1 permit 10.10.109.16 0.0.0.0 1677 tcp
outbound  1 permit 10.10.109.16 0.0.0.0 1677 udp
outbound  1 deny 0.0.0.0 0.0.0.0 1683 tcp
outbound  1 permit x.x.x.x 255.255.255.255 2323 tcp
outbound  1 permit x.x.x.x 255.255.255.255 2323 udp
```

The server group consists of a Unix server for student registration, a Novell 4.11 server with Groupwise 5.5 for campus email, and a variety of Novell 5.1 servers for the rest of campus services. Telnet is the permitted remote management protocol, however an access-list specifies only the single internal static address of the network administrator be permitted to telnet into devices.

The SMTP server has a single Pentium II 350 mhz CPU with the maximum 256 mb RAM (max), while the NOS was at Service Pack 9, with Groupwise 5.5 at SP5. The nature of Groupwise 5.5 renders antivirus software on the server ineffective. The server stores e-mails in binary files, which AV cannot scan for matching signatures. Thus, there is no AV software running on the SMTP server. Patches are applied to a server one month after they are released (the delay insures there are no retractions or problems with the released patches). However, that the SMTP server runs an outdated version of Novell proved to be part of the problem that eventually led to the Bugbear attack on the network. With only a few exceptions (see above), all outgoing ports remain open. This is because GroupWise continually ascends the TCP port ladder in opening ports to transfer e-mail, rather than using specific ports. This leads to a much more permissive policy that is closer to a “permit all except that which is specifically denied,” rather than a more stringent, recommended policy of “deny all except that which is specifically permitted.”

The print queue that was affected was housed on a Novell 4.11 server, SP9. The Compaq Proliant 4500R machine has three P133 CPUs, 512 MB RAM, and a 40-gigabyte hard drive. College has various HPJetDirect Printer devices with printer server installed. These print servers are able to handle both TCP/IP and IPX/SPX connections, the latter of which carried the exploit in this instance.

The infected workstation was a Compaq Deskpro 6266 Series with a Pentium 233 mhz CPU, 4GB hard drive, 32 MB RAM. An application server and ZENworks Application Launcher for Win32, Version 3.0 client software is used to manage distribution of applications throughout College. Thus, everyone accesses office apps through the network. Applications available include Microsoft Office 2000 (Access, Word, Excel, PowerPoint), Acrobat Reader, Novell GroupWise and Microsoft PhotoEditor. There is no specific policy in place to update patches on client machines, and no record kept of what service pack level and patches have been installed in machines. Clients are updated when other work is necessary, but the inherent flaw is that a stable client may not be updated. Clients run Command antivirus at the desktop.

In addition, the infected client was part of a legacy Organizational Unit, and was thus configured with only a single network share – to the print server. As such, when the Bugbear virus attempted to replicate, it found only the printers at College, and, fortunately, no servers or other devices. Otherwise, the damage could have been much greater.

PROTOCOL DESCRIPTION: There were several protocols that were exploited, notably MIME headers and NetBios, along with SMB, SMTP , HTTP and MAPI.

MIME Headers: Today's Internet is a far cry from the Department of Defense's original ARPAnet, which was used for communication between a handful of college and government organizations. Instead, today's Internet reaches the masses. Over the last decade, the World Wide Web has developed as a legitimate public resource. As such, is has continually pushed the envelope for presenting information to users. Today's Web is much more multimedia intensive than the ASCII-based communications less than two decades ago.

MIME, or Multipurpose Internet Mail Extensions, has helped make that the multimedia evolution possible, yet they remain an obscure (though important) detail in the study of web traffic. Perhaps because MIME are not well understood by the masses, they can be the cause of many problematic e-mails.

MIME headers are used to make non-text data appear as text to other applications, in order to facilitate file transfers. Web pages today include a variety of embedded protocols, including scripts, Java applets, graphics, sounds, video, etc. All of these sub-files must be transferred in the http-request. In the process of transferring these files, the MIME header also notifies the client requesting the document what type of files it is transferring. In this way, the client knows how to interpret word documents, executables, etc. In other words, the MIME header is a substitute or similar to the file extension found on desktop files, and, indeed, the MIME content types match up with various application file extension. [11] [For full listing of MIME content types/sub-types, see **Appendix C**.]

[11] http://www.hansenmedia.com/dict_r.asp?letter=m&ID=2137

EXAMPLE E-MAIL HEADER:

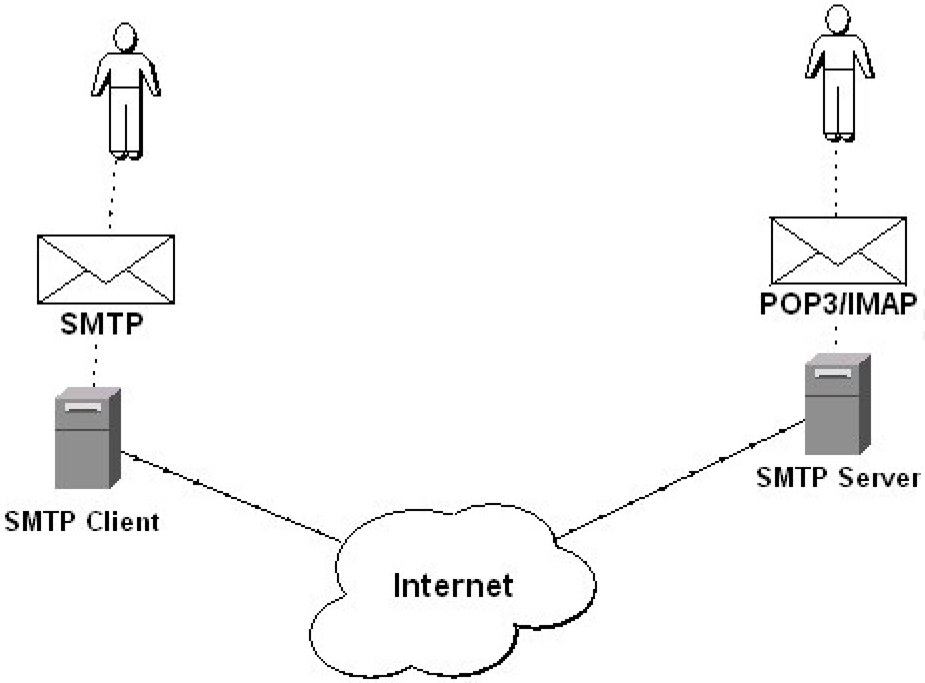
```
Received: from imo-d08.mx.aol.com
      by gwsmtp.kubrick.com; Tue, 01 Apr 2003 12:10:39 -0500
Received: from Laragiac@aol.com
      by imo-d08.mx.aol.com (mail_out_v34.21.) id m.110.21f7d481 (4238)
       for < drstrangelove@kubrick.com >; Tue, 1 Apr 2003 12:05:11 -
0500 (EST)
From: Laragiac@aol.com
Message-ID: <110.21f7d481.2bbb20c4@aol.com>
Date: Tue, 1 Apr 2003 12:05:08 EST
Subject: Re: GCIH practical (more)
To: drstrangelove@kubrick.com
MIME-Version: 1.0
Content-Type: multipart/alternative;
boundary="part1_110.21f7d481.2bbb20c4_boundary"
X-Mailer: 7.0 for Windows sub 10634
```

The MIME-Version line of the e-mail header tells the version, while the Content-type line dictates the appropriate type and subtype. This information precedes the HTML document and is part of the HTTP protocol dialog that occurs before transmission of the document itself. An incorrect MIME-version (combination of content type and subtype) can cause automatic execution of links, rather than forcing the user to click on a link or attachment. This was the initial attack methodology of the Bugbear worm.

SMTP/POP3/IMAP: E-mail was the first way Bugbear propagated through the College network environment. SMTP is an OSI Model Application layer protocol that specifies the connection-oriented TCP at the Transport layer. SMTP uses a Message Transfer Agent (MTA) and a short set of five commands (HELO, MAIL, RCPT, DATA, and QUIT) to transfer mail from one client to another. SMTP is a unique protocol that is not time-sensitive. Instead, it can be delayed at the sender site, receiver site, or in between.

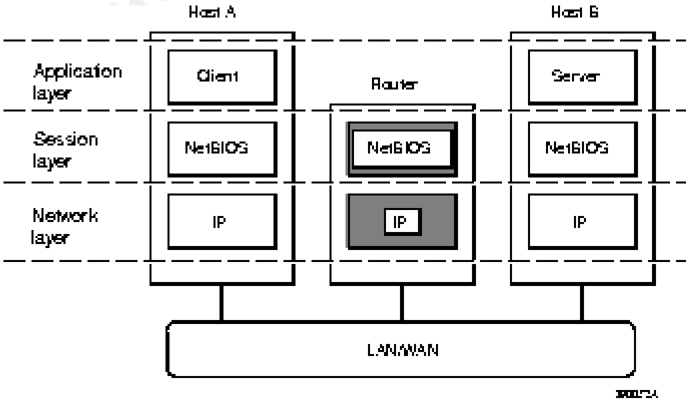
A User Agent creates the message on an SMTP client machine (workstation or server). The message is either forwarded immediately or placed in a queue to be forwarded at a predetermined time. The next SMTP device connects to the client machine and the message is forwarded into and through the cloud.

While the SMTP protocol is not time-sensitive, it does expect the receiving end to have open connections. Since this usually isn't the case with the destination machine, a mail-drop service is provided. After reaching the destination SMTP server, either POP3 or IMAP protocols may be used to connect to the server at a later point in time and either download (POP3) or view (IMAP) the message. POP3 physically moves the message to the end workstation, while IMAP allows the message to remain on the server indefinitely and permit the user to view the message from multiple locations.



Bugbear has its own SMTP engine built in, allowing it to create and forward e-mails that help it propagate through the network.

NetBIOS/NetBEUI/SMB: In addition to e-mail, Bugbear exploits network shares. In Microsoft, Simple Message Block (SMB) and Common Internet File System (CIFS) are Layer 7 (Application) protocols combine to handle the file sharing duties.



In many modern Microsoft operating systems (Windows 2000 and later), several network shares are enabled by default. Network Basic Input/Output Services (NetBIOS) is an OSI Layer 5 protocol (Session layer) that allows programs to have a common interface for sharing network and file information over various lower-level protocols.[12]

Novell utilizes Netware's NetBios as an emulator so that Windows SMB/NB services can run in a Novell environment. Similarly, Unix/Linux servers can use SAMBA to communicate with Microsoft machines.

Bugbear uses SMB/NetBIOS to jump to additional machines in the network through file shares.

TCP/IP | IPX/SPX: The Internet standard protocol suite, TCP/IP, allows connection-oriented services between machines. Internet Protocol (IP) allows machines to have logical addresses that specify both the network and host in a single 32-bit number. Information can be forwarded between hosts using this address, including hosts on different networks or network segments (i.e., the Internet). TCP, meanwhile, handles the management of the connection, including flow control and retransmission in the event of packet loss. This suite is used by the Application-layer SMTP protocol in order to forward e-mails.

IPX and SPX provide the same functionality in legacy Netware servers. IPX corresponds to IP, using an 80-bit number to specify the logical address (32-bit network ID and 48-bit MAC, thus eliminating ARP requests). SPX offers the connection-oriented services similar to TCP.

With the release of Netware 5.1, Novell networks fully support TCP/IP. Previously, a gateway server was needed for communication between operating systems.

Bugbear utilizes SMB at the Application layer and NetBIOS at the session layer. These two protocols together can be put on top of either TCP/IP (NBT) or IPX/SPX (IPXBEUI or MSIPX).

HTTP: HyperText Transfer Protocol is used to download files from the Internet. Although this virus is know for moving through e-mail, the IE vulnerability means infection could occur by visiting a website with the incorrect MIME headers, forcing IE to download the malicious program.

MAPI: Messaging Application Programmers Interface allows programmers to write source code according to a standard set of commands stored in a DLL files, and is often known as "messaging middleware". By using MAPI, it is not necessary to write a specific set of instructions for each application to interact with Microsoft Mail programs. In this case, the Bugbear virus makes MAPI calls to Outlook/Outlook Express to query the address book for information.

HOW THE EXPLOIT WORKS: As previously stated, Bugbear attacks on several fronts simultaneously. According to F-Secure, there are at least five attacks that Bugbear attempts:

[12] http://support.baynetworks.com/library/tpubs/html/router/soft1200/117358AA/B_39.HTM

1. E-mail spreading
2. System infection through Registry modification
3. Inserts keylogger
4. Kills services, including antivirus
5. Network propagation

The following sections will examine each of these in-depth.

E-Mail Spreading: Bugbear first locates the infected system's SMTP server by locating the SMTP Registry key and gleaning the appropriate information, such as address book information.

HKEY_CURRENT_USER\Software\Microsoft\Internet Account Manager\
Accounts\%Default Mail Account%
"SMTP Server"

Once this information is obtained, Bugbear can now begin wreaking havoc. It starts by generating random emails, using its own SMTP engines. Two of its three engines can create fake e-mail messages. These messages have no payload and may contain any of a number of subject lines:

- Greets!
- Get 8 FREE issues - no risk!
- Hi!
- Your News Alert
- $150 FREE Bonus!
- Re:
- Your Gift
- New bonus in your cash account
- Tools For Your Online Business
- Daily E-mailReminder
- News
- free shipping!
- its easy
- Warning!
- SCAM alert!!!
- Sponsors needed
- new reading
- CALL FOR INFORMATION!
- 25 merchants and rising
- Cows
- My eBay ads
- empty account
- Market Update Report
- click on this!

- fantastic
- wow!
- bad news
- Lost & Found
- New Contests
- Today Only
- Get a FREE gift!
- Membership Confirmation
- Report
- Please Help...
- Stats
- I need help about script!!!
- Interesting...
- Introduction
- various
- Announcement
- history screen
- Correction of errors
- Just a reminder
- Payment notices
- hmm..
- update
- Hello!

In addition, the worm can "reply" to messages already in the infected machine's inbox. The worm can spoof the From field of the e-mail by obtaining names from the first 170 e-mail addresses found on the infected machine. These names are obtained from the inbox or from files with the following extensions:

- .mmf
- .nch
- .mbx
- .eml
- .tbb
- .dbx
- .ocs

To avoid detection, the worm checks the Registry for the current user and filters this from its mass e-mail routine. The current logged-in user is found in the following key:

HKEY_CURRENT_USER\Software\Microsoft\Internet Account Manager\
Accounts\%Default Mail Account%
"SMTP Display Name"

Once the e-mails are created, the worm adds itself to the e-mail as an attachment, usually named DEFAULT.EXE. However, the worm can locate other files on the user's machine, clone that name and add one of the following extensions: .SCR, .PIF or .EXE. This results in a double-extension filename, such as PRACTICAL.TXT.SCR.

Each of Bugbear's three SMTP engines has its specialty. The first generates a content type/subtype of "application/x-msdownloaded". The second creates a type/subtype of "application/x-midi". It is this second engine that creates the message body in HTML in order to exploit the MIME vulnerability, allowing the automatic execution of the code when a user reads or previews the mail in Microsoft Outlook or Outlook Express (or Groupwise). When the user views the infected e-mail, Bugbear is automatically installed on the machine, initiating the whole process again.

System infection through Registry modification: Upon execution, the Bugbear virus immediately infects the current system. It copies itself into the Windows System directory, generating a random application name so as to mask its presence. In addition, it adds a startup key to the Registry:

```
HKEY_LOCAL_MACHINE\Software\Windows\Current\Version\RunOnce
<random_string> = \%System%\<random_filename>.exe
```

NOTE: %system% is a variable. The worm locates the System folder and copies itself to that location. By default this is C:\Windows\System (Windows 95/98/Me), C:\Winnt\System32 (Windows NT/2000), or C:\Windows\System32 (Windows XP).[13]

The worm also seeks out the Startup folder by examining the following Registry entry:

```
HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Explorer\Shell Folders Startup
```

Once Bugbear finds this key, it can generate a copy of itself into the Startup folder, using a three-character, semi-randomly generated filename. After inserting itself into the system, it can carry out its other exploits.

Inserts keylogger: Like many in the current generation of worms, Bugbear isn't satisfied with a DoS attack. Instead, it is malicious enough to plant a backdoor onto the machine, and turn on keylogging. The worm then opens port 36794 on the infected machine to allow remote connectivity to the infected machine through the open port.

Remote users have any number of options, including:

[13] http://securityresponse.symantec.com/avcenter/venc/data/w32.bugbear@mm.html

- Delete files.
- Terminate processes.
- List processes and deliver the list to the hacker.
- Copy files.
- Start processes.
- List files and deliver the list to the hacker.
- Deliver intercepted keystrokes to the hacker (in an encrypted form). This may release confidential information that typed on a computer (passwords, login details, and so on).
- Deliver the system information to the hacker in the following form:
    - User: <user name>
    - Processor: <type of processor used>
    - Windows version: <Windows version, build number>
    - Memory information: <Memory available, etc.>
    - Local drives, their types (e.g., fixed/removable/RAM disk/CD-ROM/remote), and their physical characteristics
- List network resources and their types, and deliver the list to the hacker.

A sign of actual connection through the Bugbear program may be the creation of files in the Windows temporary folder. These files include:

- ~PHGGUM.TMP
- ~EAYLNLF.TMP

The ~PHGGUM.TMP file includes a 20-character string, used by the Trojan as a session-identification number. This ID is needed for the connecting user to send information to the infected host.

Kills services, including antivirus: Perhaps the most insidious part about Bugbear is that, even if the possibility existed for detection through updated antivirus, the worm may have been able to counter that defense. This is possible because Bugbear searches for and terminates numerous processes if they are running, including the following, most of which are AV:

- _AVP32.EXE
- _AVPCC.EXE
- _AVPM.EXE
- ACKWIN32.EXE
- ANTI-TROJAN.EXE
- APVXDWIN.EXE
- AUTODOWN.EXE
- AVCONSOL.EXE
- AVE32.EXE
- AVGCTRL.EXE
- AVKSERV.EXE
- AVNT.EXE
- AVP.EXE
- AVP32.EXE
- AVPCC.EXE
- AVPDOS32.EXE
- AVPM.EXE
- AVPTC32.EXE
- AVPUPD.EXE
- AVSCHED32.EXE
- AVWIN95.EXE
- AVWUPD32.EXE
- BLACKD.EXE
- BLACKICE.EXE

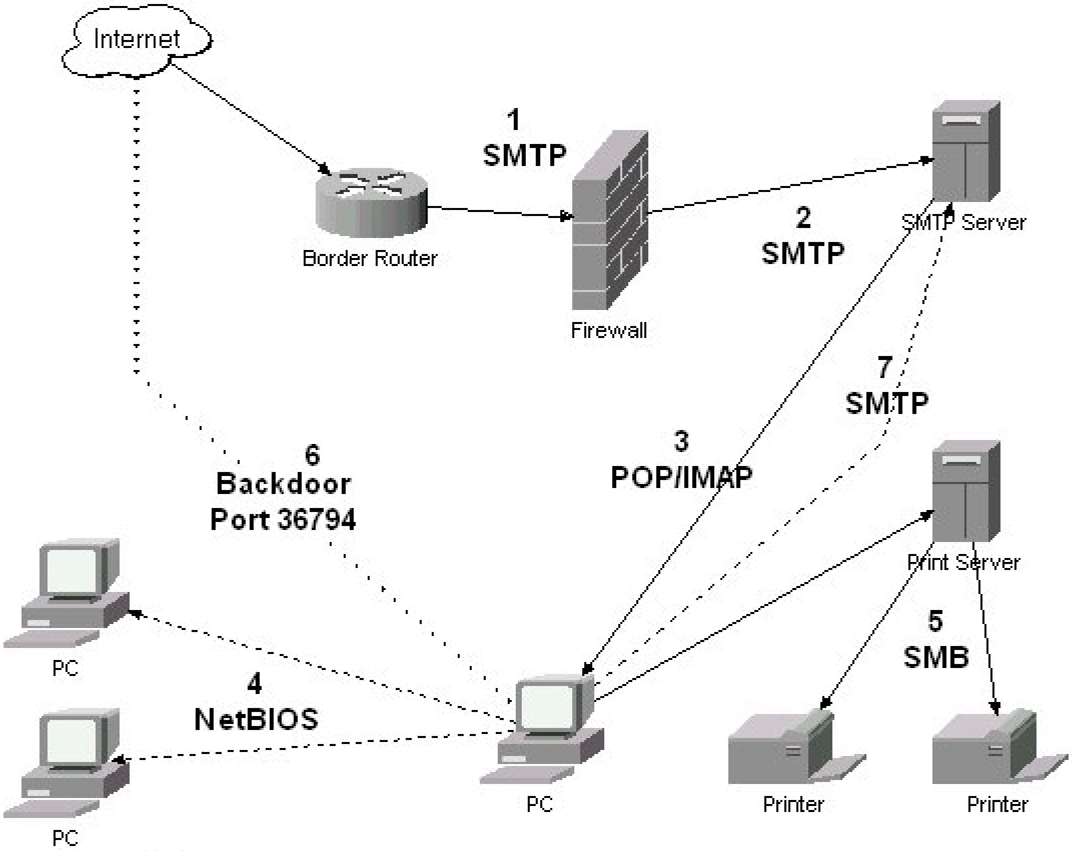
- CFIADMIN.EXE
- CFIAUDIT.EXE
- CFINET.EXE
- CFINET32.EXE
- CLAW95.EXE
- CLAW95CF.EXE
- CLEANER.EXE
- CLEANER3.EXE
- DVP95.EXE
- DVP95_0.EXE
- ECENGINE.EXE
- ESAFE.EXE
- ESPWATCH.EXE
- F-AGNT95.EXE
- F-PROT.EXE
- F-PROT95.EXE
- F-STOPW.EXE
- FINDVIRU.EXE
- FP-WIN.EXE
- FPROT.EXE
- FRW.EXE
- IAMAPP.EXE
- IAMSERV.EXE
- IBMASN.EXE
- IBMAVSP.EXE
- ICLOAD95.EXE
- ICLOADNT.EXE
- ICMON.EXE
- ICSUPP95.EXE
- ICSUPPNT.EXE
- IFACE.EXE
- IOMON98.EXE
- JEDI.EXE
- LOCKDOWN2000.EXE
- LOOKOUT.EXE
- LUALL.EXE
- MOOLIVE.EXE
- MPFTRAY.EXE
- N32SCANW.EXE
- NAVAPW32.EXE
- NAVLU32.EXE
- NAVNT.EXE
- NAVW32.EXE
- NAVWNT.EXE
- NISUM.EXE
- NMAIN.EXE
- NORMIST.EXE
- NUPGRADE.EXE
- NVC95.EXE
- OUTPOST.EXE
- PADMIN.EXE
- PAVCL.EXE
- PAVSCHED.EXE
- PAVW.EXE
- PCCWIN98.EXE
- PCFWALLICON.EXE
- PERSFW.EXE
- RAV7.EXE
- RAV7WIN.EXE
- RESCUE.EXE
- SAFEWEB.EXE
- SCAN32.EXE
- SCAN95.EXE
- SCANPM.EXE
- SCRSCAN.EXE
- SERV95.EXE
- SMC.EXE
- SPHINX.EXE
- SWEEP95.EXE
- TBSCAN.EXE
- TCA.EXE
- TDS2-98.EXE
- TDS2-NT.EXE
- VET95.EXE
- VETTRAY.EXE
- VSCAN40.EXE
- VSECOMR.EXE
- VSHWIN32.EXE
- VSSTAT.EXE
- WEBSCANX.EXE
- WFINDV32.EXE
- ZONEALARM.EXE

Network propagation: In addition to email, the worm also uses SMB/NetBIOS to spread via Microsoft network shares. To do this, it continually scans for shared resources, including drives, folders and devices. When it finds a share, it attempts to copy itself as the following:

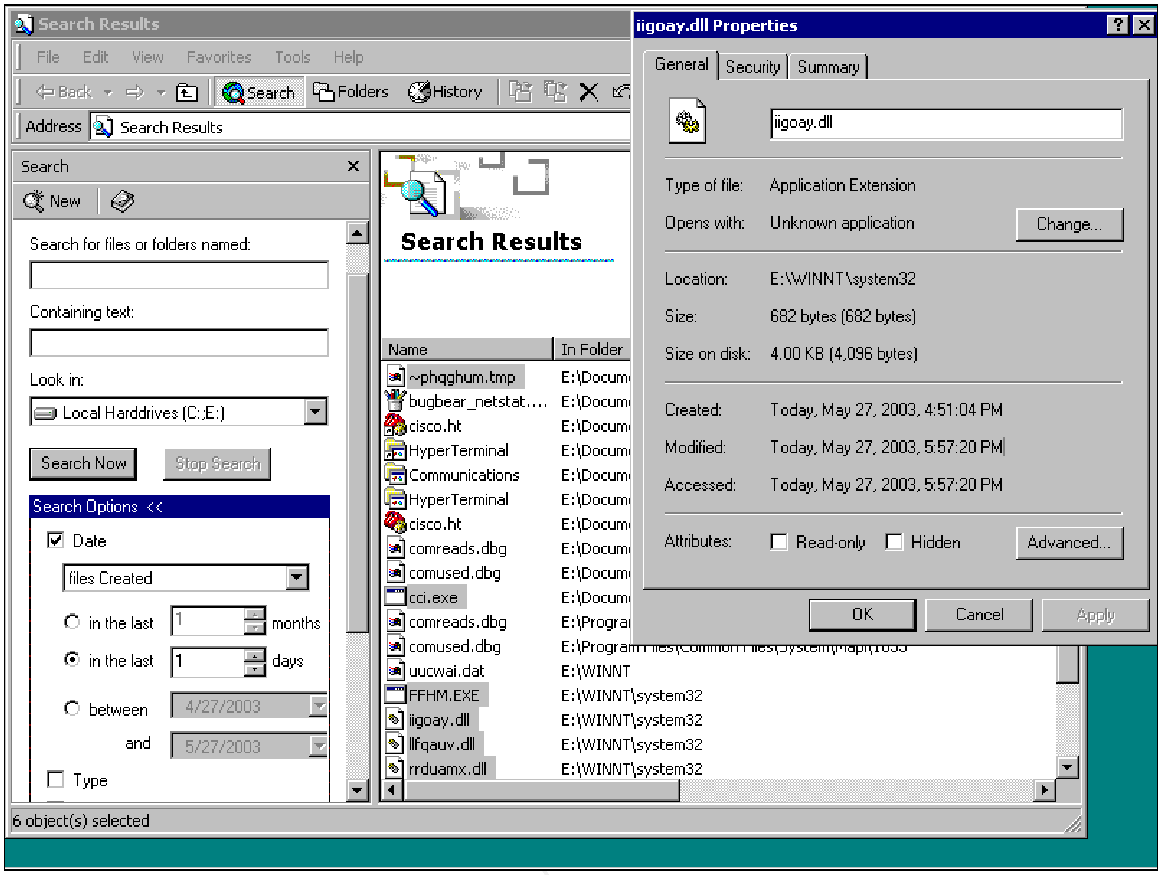
\\<shared resource name>\%Startup%\<random filename>.exe

In this instance, %Startup% is the path the Startup folder of the network share, while random filename is the same filename dropped in the Startup folder of the system the worm is trying to jump from.

DESCRIPTION AND DIAGRAM OF ATTACK: Overall, the initial attack methodology is relatively simple.



An attack on an unprotected system would most likely begin with an infected e-mail infiltrating from the outside (1). This e-mail may be deliberately sent, or part of the replication routine from another network. As standard e-mail, using SMTP, it would pass through the Internet-connecting router and through the firewall rules (2). Once on the SMTP server, the unknowing user would download (POP3) or view (IMAP) the infected e-mail (3). The MIME-header flaw would cause the e-mail to self-execute, starting the process of infiltration into the network.



Address: Search Results

Search for files or folders named:

Containing text:

Look in: Local Harddrives (C:;E:)

☑ Date

files Created

○ in the last 1 months

● in the last 1 days

○ between 4/27/2003 and 5/27/2003

□ Type

Name	In Folder
~phqghum.tmp	E:\Docum
bugbear_netstat....	E:\Docum
cisco.ht	E:\Docum
HyperTerminal	E:\Docum
Communications	E:\Docum
HyperTerminal	E:\Docum
cisco.ht	E:\Docum
comreads.dbg	E:\Docum
comused.dbg	E:\Docum
cci.exe	E:\Docum
comreads.dbg	E:\Progra
comused.dbg	E:\Program Files\Common Files\System\mapi\1033
uucwai.dat	E:\WINNT
FFHM.EXE	E:\WINNT\system32
iigoay.dll	E:\WINNT\system32
llfqauv.dll	E:\WINNT\system32
rrduamx.dll	E:\WINNT\system32

6 object(s) selected

iigoay.dll Properties

iigoay.dll

Type of file: Application Extension

Opens with: Unknown application

Location: E:\WINNT\system32

Size: 682 bytes (682 bytes)

Size on disk: 4.00 KB (4,096 bytes)

Created: Today, May 27, 2003, 4:51:04 PM

Modified: Today, May 27, 2003, 5:57:20 PM

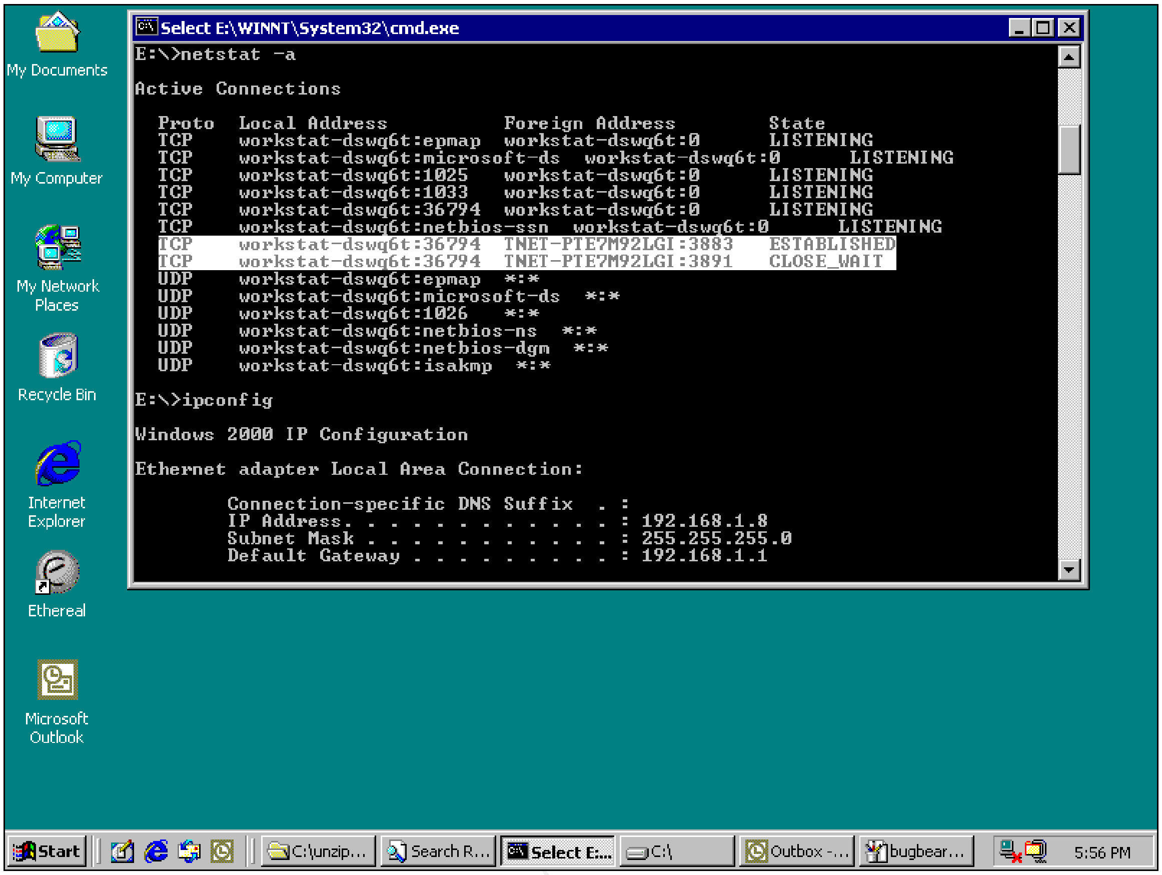
Accessed: Today, May 27, 2003, 5:57:20 PM

Attributes: □ Read-only □ Hidden

The worm would instantly install itself on the workstation, planting the described programs and files. It would then begin trying to open NetBIOS shares in order to jump to other machines internally (4).

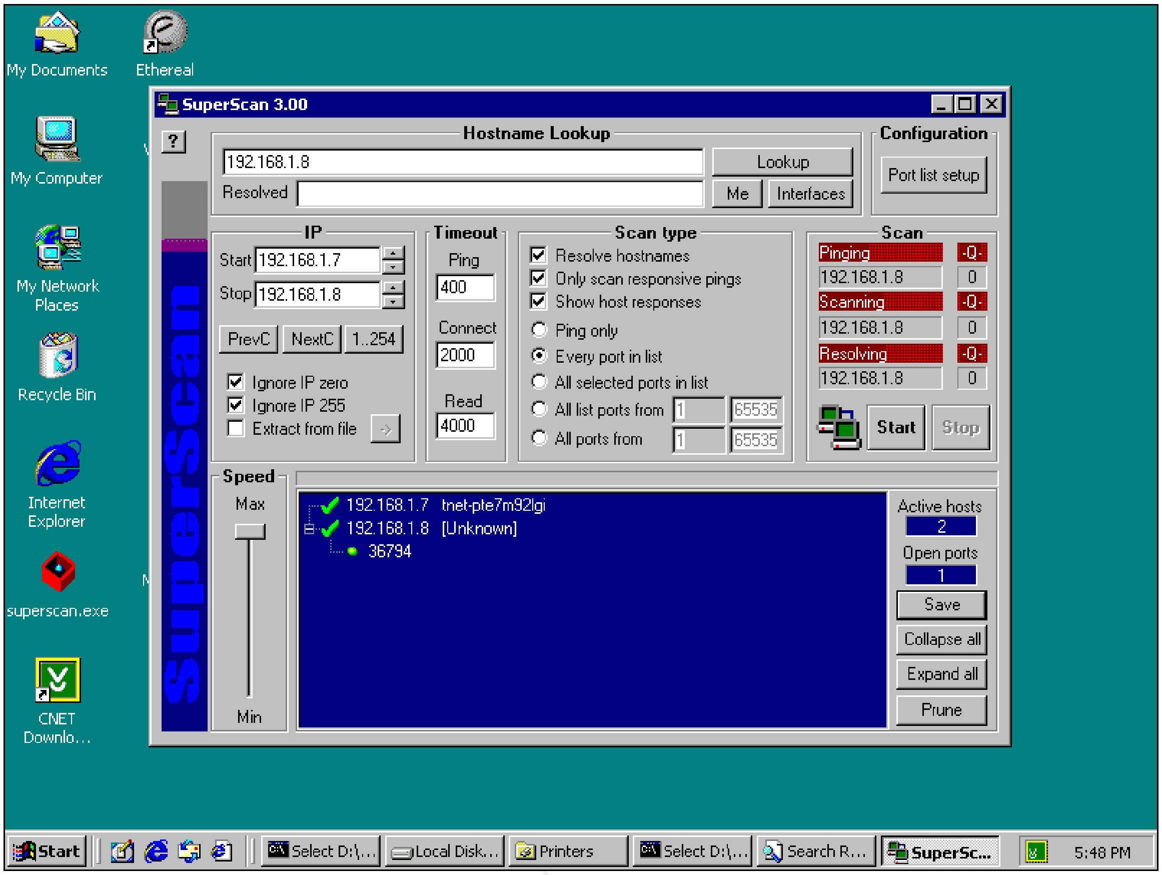
In addition, the worm would seek out printer shares by using the SMB protocol and begin printing out what appears to be garbled print jobs (5).

The worm plants a backdoor Trojan (PWS.Hooker.Trojan), which opens up port 36794 on infected machines (6). With the keylogger installed, information could be transferred to the attacker, possibly comprising sensitive files and data as well as passwords and system information. Also, with one or more infections, an attacker (either the original or someone else) could deliberately target a system for more malicious attacks by attempting to upload other programs to the infected machine(s).



```
E:\>netstat -a

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    workstat-dswq6t:epmap  workstat-dswq6t:0      LISTENING
  TCP    workstat-dswq6t:microsoft-ds  workstat-dswq6t:0      LISTENING
  TCP    workstat-dswq6t:1025   workstat-dswq6t:0      LISTENING
  TCP    workstat-dswq6t:1033   workstat-dswq6t:0      LISTENING
  TCP    workstat-dswq6t:36794  workstat-dswq6t:0      LISTENING
  TCP    workstat-dswq6t:netbios-ssn  workstat-dswq6t:0      LISTENING
  TCP    workstat-dswq6t:36794  TNET-PTE7M92LGI:3883   ESTABLISHED
  TCP    workstat-dswq6t:36794  TNET-PTE7M92LGI:3891   CLOSE_WAIT
  UDP    workstat-dswq6t:epmap  *:*
  UDP    workstat-dswq6t:microsoft-ds  *:*
  UDP    workstat-dswq6t:1026   *:*
  UDP    workstat-dswq6t:netbios-ns  *:*
  UDP    workstat-dswq6t:netbios-dgm  *:*
  UDP    workstat-dswq6t:isakmp  *:*

E:\>ipconfig

Windows 2000 IP Configuration

Ethernet adapter Local Area Connection:

        Connection-specific DNS Suffix  . :
        IP Address. . . . . . . . . . . . : 192.168.1.8
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 192.168.1.1
```

An attacker could find the machines by using a port-scanner to search for port 36,794. This could be done internally much easier, as the firewall may block a scan from the outside. [There is College policy against such behavior internally (see Appendix A). Also, in order to exploit the already-installed worm, the attacker would need the session ID stored in ~PHGGUM.TMP, meaning the attack would likely have to come from the inside for this to happen. The PIX 515 or the Cisco 2610 border router could be configured via access list to prevent outside intrusion (see **Part Three: Incident Handling Process: Lessons Learned**).] However, regardless of where the scan takes place, an attacker, upon finding 36,794 established, could attempt to escalate privileges to gain Administrator rights and/or upload malicious programs to the infected machine(s). With the disabling of Antivirus, the attack could potentially could go unnoticed indefinitely.



SuperScan 3.00

Hostname Lookup: 192.168.1.8
Resolved:

Configuration

IP
Start 192.168.1.7
Stop 192.168.1.8
☑ Ignore IP zero
☑ Ignore IP 255
☐ Extract from file

Timeout
Ping 400
Connect 2000
Read 4000

Scan type
☑ Resolve hostnames
☑ Only scan responsive pings
☑ Show host responses
○ Ping only
● Every port in list
○ All selected ports in list
○ All list ports from 1 65535
○ All ports from 1 65535

Scan
Pinging -Q-
192.168.1.8 0
Scanning -Q-
192.168.1.8 0
Resolving -Q-
192.168.1.8 0

```
✓ 192.168.1.7  tnet-pte7m92lgi
✓ 192.168.1.8  [Unknown]
     • 36794
```

Active hosts 2
Open ports 1

Finally, with the worm planted and installed, additional e-mails may be generated through Outlook or Outlook Express that continue to spread the virus (7). This would not only cause the program to be installed en masse, but, as the cycle multiplies, could sap system resources. This could eventually lead to the downing of part or the entire network.

SIGNATURE OF THE ATTACK: There are several ways to detect this attack. The most obvious is the printer behavior. A sudden up tick in the number of print jobs and reports of strange print jobs is a possible sign of infection.

The worm mostly targets Windows machines, but in integrated environments (Mac/*nix/Windows/Novell/Citrix), detection could be key, even on non-affected machines. Netminder is commercial packet analyzer for Macintosh that has been updated to include Bugbear signatures (http://www.neon.com/bugbear.html).

Also, with proper incident handling, a strain could be contained and examined with such free packet filtering tools as TCPDump/Windump, Ethereal, Procmail or Snort. Following is an example of a Snort signature:

Among the signatures that could be used:

Snort[14]:

```
alert tcp any any -> any 25 (msg:"Bugbear@MM virus in SMTP";
content:"uv+LRCQID7dIDFEECggDSLm9df8C/zSNKDBBAAoGA0AEUQ+FEN23
f7doqAT/dCQk/ xWcEQmDxCTD"; sid:900001; classtype:misc-activity; rev:1;)
```

Procmail (for Klez and Bugbear)[15]:

```
# Trap Klez (signature as of 04/26/2002)
# Trap BugBear (signature as of 10/06/2002)
#
:0
* > 50000
* ^Content-Type:.*multipart/alternative;
{
        :0 B
              * \<i?frame +src=(3D)?cid:.* height=(3D)?[0-9] +width=(3D)?[0-9]>
              * ^Content-Type:.*audio/
        * ^Content-ID:.*<
        * ^Content-Transfer-Encoding: base64
              * ^TVqQAAMAAAAEAAAA
              {
                            :0 hfi
                            * > 100000
                            | formail -A "X-Content-Security: [$HOST] NOTIFY" \
                                        -A "X-Content-Security: [$HOST] DISCARD" \
                                        -A "X-Content-Security: [$HOST] REPORT: Trapped
possible Klez worm - see
http://securityresponse.symantec.com/avcenter/venc/data/w32.klez.removal.tool.
html"

                            :0 E hfi
                            * > 50000
                            | formail -A "X-Content-Security: [$HOST] NOTIFY" \
                                        -A "X-Content-Security: [$HOST] DISCARD" \
                                        -A "X-Content-Security: [$HOST] REPORT: Trapped
possible BugBear worm - see
http://securityresponse.symantec.com/avcenter/venc/data/w32.bugbear@mm.re
moval.tool.html"

              }
```

[14] http://www.der-keiler.de/Mailing-Lists/securityfocus/focus-ids/2002-10/0035.html

[15] http://www.impsec.org/email-tools/local-rules.procmail

```
    :0 B E hfi
        * H ?? ^Subject: A( (special|very))?[ ][ ][a-z]
        * ^Content-Type:.*application/octet-stream
    * ^Content-ID:
    * ^Content-Transfer-Encoding: base64
        * ^TVqQAAMAAAAEAAAA
    | formail -A "X-Content-Security: [$HOST] NOTIFY" \
              -A "X-Content-Security: [$HOST] DISCARD" \
              -A "X-Content-Security: [$HOST] REPORT: Trapped possible Klez
worm - see
http://securityresponse.symantec.com/avcenter/venc/data/w32.klez.removal.tool.
html"
}
```

HOW TO PROTECT AGAINST IT: The most obvious and simplest way to prevent this attack is to simply PATCH YOUR SYSTEM! This attack is based on a vulnerability documented in a Microsoft Security Bulletin dated March 29, 2001, 18 months prior to the Bugbear attack. At that time, the bulletin stated that a patch was available, located at:

http://www.microsoft.com/windows/ie/download/critical/Q290108/default.asp

In addition, the patch is included in Service Pack 2 for Internet Explorer 5.01 and 5.5. Alternately, users could upgrade to IE 6, which also is not affected. As previously mentioned, there are numerous fixes for Microsoft Outlook as well, and MS Outlook SP2 and MS Outlook XP SP1 will block the attachment download as well, while MS Outlook Express 6 can also be configured to block attachments.[16] Link to patches.

CAUTION: Users should take great care before applying patches or Service Packs, or upgrading software, as unforeseen conflicts could arise. Any such task should be tested in a controlled environment before field deployment. Finally, a full backup of the system should be made before the upgrade is implemented.

Additional avoidance procedures could include the deletion of executable files off the SMTP server. Having the server strip out certain file extensions could halp avoid this and numerous other viruses and worms. It should be noted, however, that in this particular incident, this was not an option. As previously discussed, e-mail traffic cannot be deciphered on the mail server because they are stored in binary files, only through on the IMAP and POP3 connections.

[16] http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/virus/alerts/bugbear.asp

While Bugbear originally terminated antivirus processes; most have been updated to prevent this behavior. Current versions of Command, as well as others, are no longer affected by this part of the worm. Following proper defense-in-depth methodology dictates running antivirus on the server as well as workstations. Thus, as is usually the case, updated antivirus should provide protection.

In addition, after infection, there are numerous removal tools available. Check any major antivirus site for these.

There are several free and commercial IDS products that can be configured to detect the virus. Again, with proper defense-in-depth tactics, an IDS detector could be placed between the firewall and the internal router-switch. While this would not provide a defense alone, the warning it provides could alert a network administrator to the problem, allowing the IH team to go into action immediately and minimizing damage.

Use of the backdoor could be prevented at the perimeter through a simple Cisco ACL on the routers (see below) or rule on the PIX firewall could be created to block port 36794 (As of PIX Firewall Release 5.12, access lists may be used in place of Conduit and static commands). This ACL could be applied in either direction, to prevent connections in or out.

ROUTER -- *access-list 101 deny tcp any any eq 36794*

PIX -- *inbound 1 deny 0.0.0.0 0.0.0.0 tcp 36794*

A simpler way of preventing this attack is to disable File Downloads in IE's Security Zone. This would prevent the worm from executing. However, this could cause significant problems within campus for faculty and staff. One policy idea may be to prevent downloads on public machines, such as those in the Learning Resource Center.

Yet another radical approach would be to cease use of IE. Other browser options exist, though many users don't realize this. Since many virus exploits take advantage of IE, use of Opera, Mozzilla or Netscape may be looked to as an alternative. College's use of Novell is in part due to this same philosophy of avoiding Microsoft.

PART THREE

The Incident Handling Process

This part of the paper examines the specific incident at College. While the incident could be contained and eradicated, many of the recommended SANS Best Practice incident handling procedures (SANS course 4.1 Incident Handling Step-by-Step Computer Crime Investigation) were not followed. This part of the paper also seeks to offer guidance on improving the handling of incidents. In order to show the suggested Best Practices, each of the six parts of Incident Handling will be divided into two subsections, one describing the actual incidents and the second offering suggestions for future incidents.

PREPARATION: Unfortunately, there were few countermeasures in place at the time of the incidents. In fact, there is little written policy at College concerning handling of incidents (The extent of College's written policy can be found in Appendix B). Among the lacking policies: no IH team, no consideration of law enforcement, inability to properly update systems, etc. In spite of the flaws in lack of planning and formal training on Incident Handling, many other SANS Best Practices were adhered to, including isolating of system and peer notification.

Incidents: Command 4.6 antivirus software was installed on each individual machine, but this version of Command required individual users to download his/her own definitions periodically. While policy dictated once per week, the reality is most users did not update their own definitions, and technicians were generally too busy to check each individual machine on campus.

Beyond that, little else was done. Memory restrictions and OS design prevented use of antivirus on the server, as previously discussed. Previously, MIS had attempted to implement Command server edition, but conflicts with Novell Netware OS and the aging nature of the SMTP server prevented that. (See **Part Two: The Attack** above).

Although it is not written policy, it is the standard MO of MIS that no time is spent on recovery of data from virus-infected machines. They are simply nuked and rebuilt. No data is backed up or retrieved, and all data on the machine is lost. To prevent the potential loss of data in advance, a personal network drive (isolated partition on the file server) is available to many users upon request, with a standard 5 GB quota limit. This network drive can be used to back up information, however, most users are not aware of this option, or the potentially drastic consequences of infection.

The PIX Firewall and Cisco routers contain the following warning banner:

> Unauthorized use of this system is strictly prohibited. For assistance, call xxx-xxx-xxxx.

System users (there are over 500) have several avenues to report problems. According to policy "When computer equipment is in need of repair, employees should report the problem to MITS (sic), extension xxxx." Additionally, problems may be reported via e-mail. Typically, problems are reported to the Technical Services Supervisor in the MIS department, whose job is to oversee support for the campus PCs. Problems are reported to this person via e-mail or phone/voice-mail, then delegated to the appropriate technician or network administrator.

There is no specified response team or emergency team in place. The MIS department is relatively small for the size of College, and there is inferred control of system resources. The size of the department (8-10 employees) does not provide for much hierarchy. Rather, most everyone in the department has specific responsibilities to handle, although three lab support technicians are under the supervision of the TSS. All employees in the department answer to the MIS director.

Systems are implemented with a standard install. Among the attributes configured before deployment: file extensions are unhidden (in Explorer, Tools → Folder Options, Click View, under Hidden Files and folders, uncheck Hide Extensions for known file types). This would guard against the double-extension behavior, such as that exhibited by this virus, whereby a user might mistakenly click on a file labeled ***.jpg, but with a hidden extension of .exe. The resulting ***.jpg.exe would appear to be a picture, but in reality would be an executable.

As for patches, the MIS department makes a valiant attempt to keep machines up to date, but as is the case at most state institutions, there are far too few bodies to go around. Although one could argue that security is the most important thing that can be done, resources – both physical and monetary, often restrict the reality of implementation. In an ideal world, all machines on campus would be patched on an ongoing basis, perhaps weekly, perhaps monthly, as policy dictates. As previously stated, the vulnerability that Bugbear exploited was 18-months old. Thus proper service packs could have prevented it.

Currently, the network servers are backed up once a week using Veritas 7.x. These backups are kept on a five-week rotation. File responsibility is left to the individual user.

Recommendations: There are several areas that have the potential for improvement, including a more comprehensive policy, warning banners security templates, development of an Incident Handling team, logging, and improved organizational communications about incidents.

Policy: Perhaps most important, a written policy, whatever that might or might not include, should be established. This policy should include the rights (or lack of rights) of users of the systems, along with specific penalties, up to and including termination of employment (for employees) or suspension/expulsion (for students). This policy should also provide for the option of notifying law enforcement. While much of this is assumed, having employees sign this provides a mental deterrent, in addition to any physical deterrents created by hardening the systems. This policy statement could be included in all new-employee packets, and should be distributed by MIS for all current users. A grace period could be provided, with users locked out of accounts after the expiration of the grace period. While this recommendation does not specifically address the Bugbear virus, it is, nevertheless, the foundation on which all Incident Handling actions are built.

Banners: While the current warning banners may be adequate, it would be better to plainly state that monitoring and recording of the system can take place. This provides the Incident Handling team with a legal foundation to monitor the attacker's activities. Also, legal experts should probably evaluate the warning banners to ensure effectiveness. Current law is divided and ambiguous on the implied rights of users. It is better to plainly state user rights (or lack thereof) to better counter any potential legal loopholes.

One example might be:

> This is College computer system. This resource, including all related equipment, networks and network devices, are provided for authorized College use. College computer systems may be monitored for all lawful purposes, including to ensure authorized use, for management of the system, to facilitate protection against unauthorized access and to verify security procedures and operational procedures. The monitoring on this system may include audits by authorized College personnel to test or verify the validity, security and survivability of this system. During monitoring information may be examined, recorded, copied and used for authorized purposes. All information placed on or sent to this system may be subject to such monitoring procedures. Use of this College computer system, authorized or unauthorized, constitutes consent to this policy and the policies and procedures set forth by College. Evidence of unauthorized use collected during monitoring may be used for criminal prosecution by University staff, legal counsel and law enforcement agencies.

No warning banners are posted on the servers or client machines, even though these were the targets of the attack. Banners can be set in Windows by editing the following Registry key in Windows NT/2000/XP:

HKEY_LOCAL_MACHINE
SOFTWARE\Microsoft\WindowsNT\CurrentVersion\Winlogon\LegalNotice
Text

For the Netware servers, an advanced version of the Login.exe program is available from Novell.[17] After downloading this file, the server administrator can create a warning banner in a LOGIN.txt file. The server will then display the contents of this file at login.

Security Templates and Benchmarks: The Center for Internet Security has developed the Gold Standard template for Windows 2000. Simply put, this template saves a lot of the grunt work for putting together basic security measures on Windows machines, by predefining a standard commonly agreed to by many of the top security professionals in the industry. The template is held in such high regard, it is mandatory implementation in many government organizations, including the Department of Justice. Implementation of this across campus (where possible) would also significantly reduce threats. Although not all machines use Windows 2000 operating system, the documented key changes in the template could be created in a Windows XP template and likewise applied. Other benchmarks are also available at www.cisecurity.org.

Incident Handling team: Careful consideration should be given to developing a full-fledged response team. In the event of a more serious or prolonged attack, this would provide additional human resources and perspectives on the incident. Members of the response team do not necessarily need to be limited to MIS departmental employees. Rather, this team could, and possibly should include outside help, including law enforcement and technology instructors. This team should be properly trained and should practice its IH skills through periodic exercises. A meeting facility and established budget to cover expenses (supplies, compensation, consulting fees, etc.) should be included as well. The team should have access to a static supply cache (additional hard drives, tools, boot disks, etc.).

The internal cost may be prohibitively expensive, given the few occurrences of serious incidents. On the other hand, the use of outside help is admittedly a security concern as well, especially the need of the Incident Handling team to have passwords. Possible solutions could include less-serious incidents (such as viruses) are handled in-house, as they currently are while more serious incidents are handled by a specified consulting team. The setting of specific accounts for Incident Handlers, with login information (username/password) made available to outside team members only after entering into Incident Mode. Afterward, these accounts could be reset. If this option is employed, passwords should be stored in a secure, but accessible place (i.e. a small safe, a locked drawer, etc.), possibly with some sort of encryption software. Furthermore, this information

[17] http://www.novell.com/coolsolutions/tools/1144.html

could be stored on a floppy disk, in a password-protected file. The implementation of a TACACS+ security server could also centralize all passwords and access levels, with the Incident Handling process being just one of the beneficiaries of implementation.

Organizational Communications: Notification of system users is a key issue as well, since the situation may arise where users may still need access to compromised systems, and should be aware of this fact. Alternately, situations may arise where a compromised system may need to be taken down. Again, proper notification is key to getting users to properly log off the system. As an example, consider that users are notified when other networks in the state college system are taken off-line. Likewise, policy should dictate when/who/what/how to notify outsiders, i.e. other colleges and state institutions, etc.

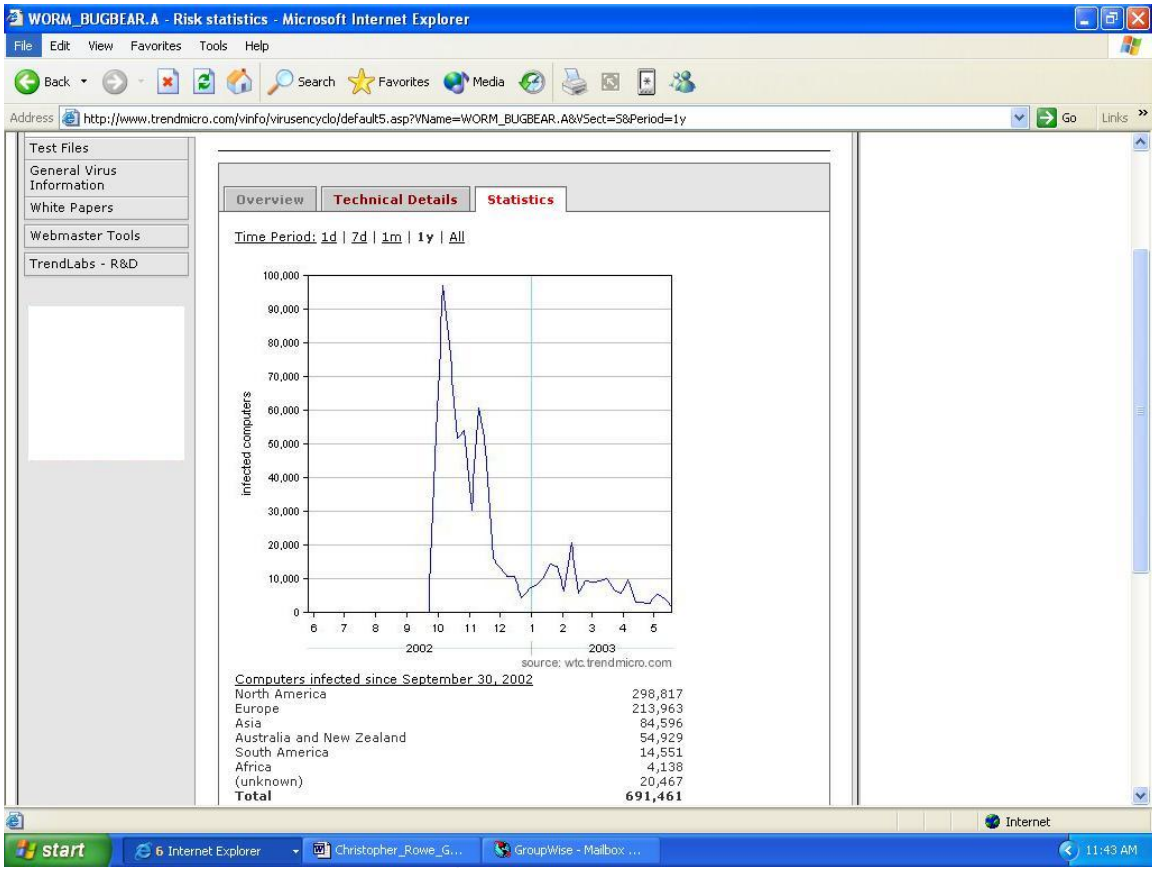
Law-enforcement: College has an on-campus police force, as well as a county sheriff precinct. Thus, consideration should be given to proper interaction with law-enforcement. On-campus crime is not limited to parking lot break-ins, assaults, and classroom equipment theft. Cyber crime's stealth nature dictates having specialized and pre-determined procedures in place in order to preserve evidence. Big issues and hypothetical issues and policy on handling these issues should be discussed up front. Should law-enforcement be notified about a virus? While the instinct is to disregard a virus as a low-level threat, no decision should be made without at least consideration of how to handle it. Who will be notified? What will be done? What is considered a high-level threat?

Maintenance: Furthermore, proper network maintenance also helps in the event of cyber crime. Having proper backups, and the policy of quick restoration, can get a network back up and running in minutes or hours, not days. In this instance, there were backups of the SMTP server available, if needed, however, no backups of individual machines existed.

Logging – At current time, the server resources prevent proper logging. By logging such things as successful and failed logins, errors, and odd packets can give advance warning that an attack may be impending. Currently, the PIX uses syslog level 5. This same philosophy should carry over to the servers.

Quite simply, a routine procedure should be implemented to deal with many of these threats. It is likely only with proper preparation that a serious threat can be averted. Preemptive actions such as developing warning banners on all systems and deciding on whom and how to respond to incidents should be part of the planning process. In the end, the MIS director and College's president (and/or Board of Trustees) should approve this policy.

IDENTIFICATION: Bugbear first appeared in the wild September 30, 2002. The threat has mostly subsided, but the virus is still in the wild, infecting thousands of machines daily. Additionally, the virus' troubling behavior of turning off AV software has almost certainly gone undetected by some and left an untold number of machines at risk for numerous other attacks.



source: wtc.trendmicro.com

Computers infected since September 30, 2002	
North America	298,817
Europe	213,963
Asia	84,596
Australia and New Zealand	54,929
South America	14,551
Africa	4,138
(unknown)	20,467
Total	**691,461**

Incidents: Four days after the initial appearance of the Bugbear virus, it hit College. As College opened for business that morning, MIS quickly began receiving reports of strange network activities. Indeed, it was the garbled printer outputs that tipped off MIS that something was amiss, and led to quick identification.

The first step was to check Novell's printer queues with Pconsole. Sure enough, numerous print jobs were found. These were immediately deleted. By examining the timestamps of each job and the machine that outputted it, the infected workstation was located.

At this point, it was still unclear whether this was a virus, or another problem. The system's user was contacted and told to shut the machine down. (It is usually better to simply "pull the plug." A hard shutdown prevents the attacking program from determining it is being handled and deleting itself). The system was pulled from the network and brought to the MIS office. Once the infected machine was retrieved, the next step was to identify the problem.

MIS rebooted the machine with it attached to a faux network. This is a cautionary measure that usually fools a virus into keeping its networking components active and again avoids tipping off the program that it is has been discovered. An examination of the machine showed numerous strange processes running, but when these processes were terminated, they immediately restarted.

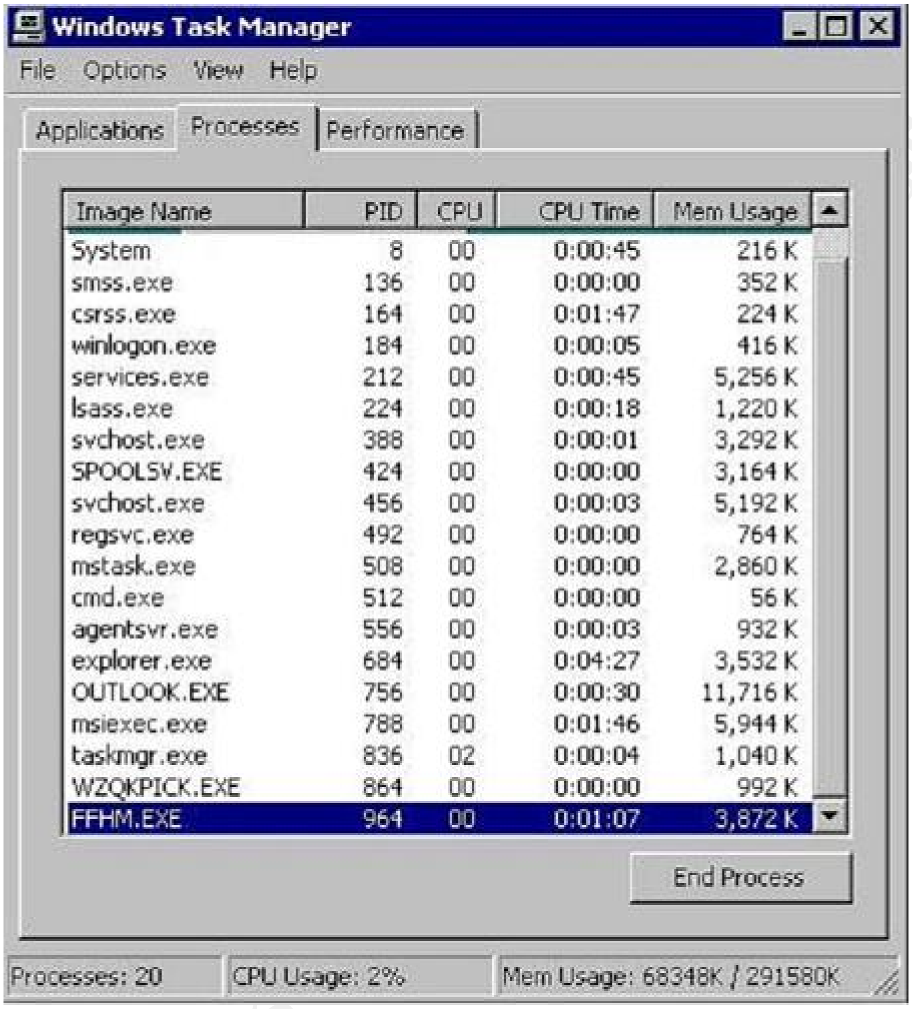


Image Name	PID	CPU	CPU Time	Mem Usage
System	8	00	0:00:45	216 K
smss.exe	136	00	0:00:00	352 K
csrss.exe	164	00	0:01:47	224 K
winlogon.exe	184	00	0:00:05	416 K
services.exe	212	00	0:00:45	5,256 K
lsass.exe	224	00	0:00:18	1,220 K
svchost.exe	388	00	0:00:01	3,292 K
SPOOLSV.EXE	424	00	0:00:00	3,164 K
svchost.exe	456	00	0:00:03	5,192 K
regsvc.exe	492	00	0:00:00	764 K
mstask.exe	508	00	0:00:00	2,860 K
cmd.exe	512	00	0:00:00	56 K
agentsvr.exe	556	00	0:00:03	932 K
explorer.exe	684	00	0:04:27	3,532 K
OUTLOOK.EXE	756	00	0:00:30	11,716 K
msiexec.exe	788	00	0:01:46	5,944 K
taskmgr.exe	836	02	0:00:04	1,040 K
WZQKPICK.EXE	864	00	0:00:00	992 K
FFHM.EXE	964	00	0:01:07	3,872 K

Processes: 20 CPU Usage: 2% Mem Usage: 68348K / 291580K

A survey of several antivirus sites did not yield any information. College had received an e-mail from the state office earlier in the morning, but the e-mail was only skimmed, and not read thoroughly. This e-mail was a forwarded notification from IBM Managed Security Services, Security Advisory Alert # MSS-SVA-E01-2002:001.1 (see **Appendix B**), and had been issued by IBM the previous evening. The primary network administrator continued searching the Internet for information, but did not find anything. After about four hours of searching, a second systems administrator reviewed the forwarded e-mail from the state again and noticed the comment regarding printer behavior. Now recognizing the infection, the primary handler in this case went into Containment mode.

The second incident was much easier to identify. Reports of printer problems similar to the first incident came in on the morning of Oct. 14 as the staff reported to work. A quick check of the Pconsole, and again the queue was flooded. The host was identified in the same manner, the print jobs deleted. Again, the employee was contacted and told to shut down the system. Again, the system was retrieved and brought to MIS, attached to a dummy network.

Recommendations: While the research ability of MIS is generally laudable (the overlooking of the e-mail details notwithstanding), additional tools can be used for identification. In addition to network firewalls (routers, PIX), consideration could be given to installing personal firewalls such as Black Ice on the desktop, although again cost could be a concern. Other means of identification could be to include an IDS device in the perimeter defense and/or on individual network segments. This could be hardware (Cisco IDS) or software (Snort or freeware firewall such as Kerio) depending on cost. Either of these solutions could provide an additional layer of defense. While an IDS will not stop the attack, it could provide notification that an attack is underway. Additionally, while Syslog is sometimes used with the PIX, the degradation in performance is a concern. As such, an upgraded firewall could allow adequate logging that admins could analyze for suspicious behavior. Tools such as MD5 and Tripwire could be used to check for file integrity. Finally, logging at the server AND desktop level (login events, etc.) could provide an additional means of detection.

While all of these may require additional human resources, the key (to be addressed later) is training users to recognize potential problems and report them to MIS. This information should be reported to one person designated as the assigned primary handler (this may/may not be network administrator). This primary handler would then make the call on what course of action (discussed in the preparation sub-section above) to take. This recommendation may cause concern that errors will occur and users will report activity that is not actually incidents. However, you can never be too cautious, and erroneous information should be treated as an opportunity to practice incident handling skills.

CONTAINMENT: With the workstation removed from the network, MIS considered other options for containment. However it was decided that shutting down the network was not a practical solution. With the infected host removed, the worm did not appear to be spreading to other machines (no other print jobs were restarting, no other reports of odd network behavior). Thus, the inconvenience of shutting down the SMTP server, which is the primary means of communication at College (a recent technology survey put usage at 85 percent), would cause more problems than it would offset with what was, at this point, considered a near-zero risk of the worm spreading. However, news of the infection was brought to the school’s attention via a mass e-mail. The e-mail stated an infection had been found and reminded people to not open e-mail attachments. At the time, it was not known the virus could potentially self-execute.

The now-isolated machine was rebooted in the MIS office on the isolated LAN connection. The machine had only one network share, and that was the printer share. Thus, the infection did not spread to other clients. After the identified machine was removed, Pconsole was again checked for rogue print jobs. In addition, there were no other reports of problems, leading the MIS technicians to believe that they had quarantined the infection with the removal of the machine.

For the second incident, a similar procedure was followed. Once again, after identification, the host was removed from the network and brought to MIS. The host was reformatted, OS re-installed along with updated patches and virus definitions via floppy. The machine was returned to service later that day.

Recommendations: While the course of action may have been acceptable for this level of threat, higher threats deserve more serious consideration. It is possible that the network will need to be shut down. This is a judgment call, so consideration should be given to this possibility beforehand. Also, detaching the machine from the network, but reconnecting on an isolated LAN is excellent treatment of an incident. The box also could possibly be handled with a minimal OS burned onto a CD or floppy disk. Again, this could prevent the incident handler from accidentally or intentionally manipulating the data on the disk. In the event of a system compromise, all passwords on affected systems should be changed immediately.

ERADICATION: The SANS Institute Track 4 course guide dictates, "Viruses … are pretty easy to deal with after the anti-virus companies have analyzed them. You just let the software clean up the problem … ".[18]

Incidents: While that is the case for most viruses, zero-day exploits, or even fast-moving viruses that don't yet have a removal tool are trickier. While steps are now in place for manual and automated removal of the Bugbear virus, there seemed to be only one guarantee on Oct. 3 … low-level format, re-install and get the machine back into service again. Note that a low-level format was done to ensure the remnants of the virus did not remain. A standard format command does not actually delete the files, only the references, so it could be possible for the virus to be recovered or still be lurking in the hard drive. While the removal tools existed, they were not common knowledge since the worm was just days old.

When the second incident hit 11 days later, the circumstances were nearly identical. Identification was much easier. However, despite the availability of removal tools from AV vendors, the infected machine was reformatted and the OS reinstalled. Essentially, MIS was unaware of the availability of removal tools, and thus dealt with the issue in the same manner it had the first time.

[18] SANS Course Guide 4.1: Incident Handling Step-by-step and Computer Crime Investigation. P. 99

Recommendations: Nuking a machine and getting it back up and running is usually the best course of action in order to minimize downtime, and when it comes to new viruses and zero-day exploits, this may be the only solution. However, part of this process could include making copies of the infected machines' drives in order to continue the IH process outside of the network. Making bit-by-bit copies allows the IH to have additional copies of the incident to document, and keep the original drive as evidence should law enforcement need to be contacted or the incident goes to court. In this case, a duplicate of the infected machine could have led to a more in-depth analysis of the worm (before now), possibly revealing flaws in the network architecture and/or better means of identifying other potential attacks to the system. While Netware attacks are not common, they can occur. And, as is the case in this incident, the behavior of integrated systems poses some unique concerns. All of which is to say, there is no such thing as a simple virus. Every incident should be taken with the utmost seriousness.

RECOVERY: Incidents: Both incidents were handled and recovered in similar ways. In both cases, the machine was formatted and the operating system was reinstalled, along with all other appropriate software. All current service packs and patches were downloaded and installed. The new virus definitions were copied onto a floppy and then copied onto the repaired system in both instances. In neither case was data on the machine preserved; as previously discussed, it is not standard operating procedure to recover files from virus-infected machines. Fortunately, in neither case was any sensitive data (i.e. grades, evaluations, planning units) stored locally. All critical documents and data were stored on network servers.

Once the system was completely restored, an AV scan based on the new definitions was performed and two different technicians tested the machine. Since the test showed no new signs of infection, the client was put back into production later that day. The Pconsole was checked ensure that no new garbled print jobs were being generated.

Recommendations: The recovery process includes verification that there is not still a threat; that is, verify operations, run tests and baseline, then have others retest for best results. This procedure was followed, including the reviewing of the machine by two different technicians. Once again, however, little else was done besides manage the immediate crises. A few notes were jotted down, and the "Trouble Ticket" was filed with the TSS. The worm program was not saved to test for future reference. No scanning was done to ensure there weren't rogue infiltrations in the network. No research on the worm was done to check for other effects. This led to what could be considered the unnecessarily harsh handling of the second worm.

LESSONS LEARNED: The lessons learned here are two-fold. First, the handling of the second incident showed adaptation from the first, as well as flaws by overlooking certain details. However, lessons learned should examine the procedure as much as the technical knowledge gained.

Incidents: In technical terms, the discovery and identification of the virus the first time led to a quicker reaction time after the second occurrence. The second incident was quickly identified, and by this point, proper tools were available to handle it and had already been distributed by the antivirus sites. However, better communication and documentation could help prevent a scenario in the future where unnecessary steps such as what happened in these instances are taken.

After the second infiltration on Oct. 12, after which, a removal tool became available, the scanner and removal tool were made available to each machine through Netware Client Applications. To date, scanners and removal tools for Bugbear and Klez are still available to all users logged in to the network through ZENworks.

The Command Antivirus has been upgraded to version 4.80e for Netware. This version has the capability to automatically download current virus definitions, and is set to check once per week for new signatures. This will hopefully prevent situations similar to this where the same virus is able to repeatedly infect the network.

MIS has learned its lesson about the server, as well. After much delay, the department is finally in the process of upgrading to the latest version of Novell. The new SMTP server will include a Pentium 4 2.8 ghz CPU, 300 GB hard drive and 2 GB of RAM. Additionally it will run Novell 5.1 NOS along with GroupWise 6.5. This will allow Command antivirus to be implemented in conjunction with GroupWise on the server.

The MIS director is also evaluating network redesigns to include VPNs. Using encryption could greatly reduce the potential from outside threat. Protocols such as telnet and HTTP could be replaced with SSH and HTTPS. Although not specific to stopping Bugbear or other viruses, the numbers of viral incidents in the past year have been cause for re-evaluation.

Again, MIS has realized the era we are in. Funding has been approved for the network administrator to attend CISSP training. The MIS director has indicated an interest in having all employees in the department take security classes within College. The hope by everyone is that a realization of the vast number of exploits will make people more cautious about their use of and work on the College network.

Recommendations: Ironically, despite not following recommended procedures, much can be learned from this incident. Indeed, mistakes are a part of life, but they can serve as a forum for us to learn from. Such is the case here.

As noted throughout the previous steps, there are many things that could be done different.

First, a review of many of the recommendations from the previous steps:

- *Policy* – Establish a written policy and have all users sign it. Should include notification that users should not expect privacy when using system.
- *Banners* – Improve warning banners to notify of monitoring.
- *Security Templates and Benchmarks* – Develop appropriate benchmarks and deploy security templates to help lockdown individual client machines.
- *Incident Handling Team* – Assemble a full-blown IH team. Possibly have a two-stage team, with only the core deployed for smaller incidents, such as this one.
- *Organizational Communication* – Improve notification of end users where necessary. Develop policy for notification of peer networks.
- *Law-enforcement* – Develop policy of when/if to notify law-enforcement. Coordinate with on-campus law enforcement members.
- *Maintenance* – Help end users develop skills to check for updates and help with deployment of patches.
- *Additional tools* – Consider additional tools that can be used for Incident Handling, including sniffers, IDSes, file-integrity checkers, etc. Have jump kits ready.
- *Containment* – To insure integrity of above tools, and of operating systems, have bootable CDs available. Also, consider deployment of personal firewalls at the desktop level for additional notification and defense.
- *Incident Handling procedures* – Develop standard procedures for recognition, notification and handling incidents. Document process of incident handling for possible-law enforcement purposes.
- *Verification* – Have at least two technicians handle any incident and certify a machine(s) before returned to production. Develop routine to certifying the network once threat is averted.

Other considerations:

Should more of a priority be placed on implementation of service packs and patches? While MIS may be shorthanded, perhaps end users could be trained to check for this information. Microsoft's hfnetchk.exe tool (available as part of the Microsoft Baseline Security Analyzer) can identify needed patches. College already has a series of professional development workshops that cover a broad range of topics. One thought would be to design a lunch-and-learn session around this type of training. Additionally, end users should be educated about suspicious emails and attachments. While it used to be simple, that is, don't open the attachment, that is no longer the case. The Bugbear worm serves notice that there will always be a threat. That puts added emphasis on recognizing these threats before they turn into exploits.

While the threat of Bugbear has subsided, there are still numerous other viruses running rampant, and the scenario described in this paper could play out on any given day at College. A comprehensive review of the network and policies about its use could lead to a more secure environment. At the very least, a system for specifically documenting incidents could be employed. All other aspects of MIS are documented, including requests for equipment repair, new user accounts, employee termination procedures, etc. That incidents are not handled with strict techniques reveals the lack of realization (in the past) of the threats to the system, both internal and external.

Other considerations should be given to creating a complete defense-in-depth model, with a DMZ and IDSes on every segment, and with systems locked down tighter. An IDS at the front of the network could help filter external attacks. Only the servers and (only recently) the firewall employ logs. Log files could be greatly expanded and training in review of these could lead to early recognition of incidents. Consideration could be given to providing specialized training to someone in the department, or taking advantage of other resources in the college to help explore the possibility of these changes.

Finally, training is key. Unfortunately, most vendors don't include basic security practices in their training. This is unfortunate, since many (not all) exploits can be avoided with some basic practices. Currently, a Curriculum Improvement Project is reviewing all IT training in all related state institutions, and it has been suggested that basic security skills be implemented into all aspects of IT training, from user to server administration to network support. The same holds true for current admins. Much of the pitfalls come simply from ignorance of the type of exploits available.

CONCLUSION

Perhaps to some, it seems overkill to discuss Incident Handling and viruses. The reality, however, is that virus infection is as common and normal to us as surfing the Internet itself. Unfortunately, it is that mentality -- the idea that viruses are petty nuisances, small scabs that eventually go away – that will lead to much more serious breaches of security in networks everywhere. As already documented in this paper, the newest viruses are becoming increasingly insidious and malicious. They are providing more exploits than ever before.

When it broke loose in the wild, Bugbear was a new breed of worms, noted for its backdoors and keyloggers, and raised many eyebrows with the ability to disable AV software. Yet, this will quickly become the norm. Furthermore, polymorphing viruses threaten to completely undermine the tried-and-true way of detecting viruses through established signatures. As evidence, see the Part One: Variants above, as Bugbear.B, similar to the first Bugbear but polymorphing, emerged on the same morning this paper was submitted. It is not known yet what damage the new worm will do, but many of the same procedures discussed herein apply to new attack. The continuing and growing prevalence of computers and the Internet in our daily lives offer both a rich opportunity and frightening reality of the dangers that lie ahead.

Thus it is NOT overkill to mandate proper policies when “only” handling viruses. As the most common threat, they can make for good practice for future, more arcane and high-level threats.

Fortunately, this incident did not cause major problems, although the nature of the worm could have been potentially disastrous given Bugbear’s ability to spread. Even standardizing on Novell is not a 100 percent guarantee, as the MIME-exploit shows. And in a heterogeneous environment, one that in this instance, utilizes all four major operating systems, deployments (Windows/Unix/Novell/Macintosh) in some form or another, are going to be susceptible to attacks.

But it all starts with good policy and proper planning. That planning should be to follow the six-step process for Incident Handling, and take every threat seriously. Many of the security problems of the world come from simple ignorance of users, administrators and supervisors. In an interview for the SANSFIRE 2003 catalog, Eric Cole, renowned security trainer and author of *Hackers Beware* claims the solutions are simpler than we realize.

“Network Security has little to do with spending money and installing devices … but has everything to do with knowing the organization’s critical information and what risks exist related to that information,” he says.

It is the last word that is the key. Information.

“When it comes to network security, knowledge is power and ignorance is deadly.”

He should know. So should everyone else!

Appendix A

Telecommunications/Internet Use policy from COLLEGE Management Manual:

2.035 Telecommunications

2.0351 Internet Use-General

Internet services are provided for COLLEGE students to support their educational needs and for COLLEGE faculty and staff to support their professional activities. All COLLEGE users are responsible for using the Internet in an effective, efficient, ethical and lawful manner. Internet access is a privilege, not a right, and as such, can be withdrawn from those who use it irresponsibly.

Procedure:

1. Acceptance of the Internet configuration and software files constitutes agreement with the policies outlined, and understanding of appropriate and inappropriate use.

2. Users have access to a wide variety of information via COLLEGE Internet services. The availability of such information does not imply that COLLEGE approves or endorses its content. Additionally, there is no guarantee of the validity or accuracy of information accessed.

3. Users of COLLEGE Internet services should be aware that files and electronic mail are not completely private. Efforts are made to maintain the reasonable privacy of users' files on COLLEGE's local servers. However, users should be aware that COLLEGE would be compelled to share any files requested as a result of legal process or as otherwise required by law.

4. Following is a list of unauthorized activities. It is not exhaustive; users should not assume that any system use not specifically excluded is authorized or that it will be treated as such. If there is a question about whether a specific use would be permitted, it should be referred to the Internet Administrator, who will take it to the Internet Committee if necessary.

a. COLLEGE accounts are to be used solely by COLLEGE faculty, staff, and students. Employees and students may not give other persons including relatives or friends access to their accounts.

b. Individuals may not conduct activities for personal gain via COLLEGE Internet services. This includes advertising personal services, selling, soliciting jobs, or any other activities whose purpose is to generate revenue for an organization or for the individual's personal gain.

c. Activities which interfere with the ability of other users to make effective use of COLLEGE computer services are prohibited. Such activities include but are not necessarily limited to harassing or threatening other users; attempting to steal passwords or other restricted information; attempting to crash the system; attempting to gain access to directories or files for which a user is not authorized; or actions which adversely affect the performance of the computer system. Users are expected to abide by the rules of other networks that they may access via the Internet.

d. Copying, providing, receiving, or using copyrighted material in violation of licensing agreements is prohibited. Page creators using copyrighted material must obtain written permission documentation from the copyright holder, and complete a web page agreement (see forms) to be filed with the server administrator. If student photographs are used, written releases must be completed and filed.

Use of Internet services for any illegal activity will result in loss of access without prior notice. Such activities include but are not limited to computer hacking or fraud. Legal action may also be taken.

f. Any software which can be classified as "push technology."

5. COLLEGE reserves the right to examine user files, accounting information, and backups generated by use of the computing system. System administrators have the authorization and ability to monitor any user's files if there is a performance reason to do so or a specific reason to believe that a user has engaged or is engaging in unauthorized activities.

a. When a process is consuming excessive system resources or degrading system response it may be terminated, or its priority may be altered without notice.

b. Generally, a reasonable attempt will be made to notify users of a first offense. Serious or repeated offenses will result in immediate suspension or cancellation of access depending on the severity of the offense.

6. Violations will be reviewed by the Internet Committee, which may refer them for further actions to appropriate COLLEGE authorities who may impose disciplinary actions as specified in the policies of the college.

7. Individual departments may have additional rules and regulations pertaining to Internet use in their areas. Users are also expected to abide by these additional rules.

Approved 6/20/96
Rev. 7/12/99

Appendix B

E-mail notification from state MIS department:

From: "John Doe" <John.Doe@mail.net>
To: < John.Doe@mail.net >
Date: 10/3/02 7:48AM
Subject: WORM W32Bugbear@MM (UPDATE1)

Please read this one there are some excellent steps to train our users to avoid virus's. The original alert is at our FTP site ftp://ftp.its.state.nc.us/Security/2002/ Please frequent that site and bookmark it. Note that it contains alerts not sent out via email. Check to see what other areas listed are of importance to you and check them daily. Thanks for all you do - John

~~~~~~
E-mail correspondence to and from this address may
be subject to the North Carolina Public Records Law
and may be disclosed to third parties.
~~~~~~

-----Original Message-----
From: IBM MSS Advisory Service [mailto:advisory@us.ibm.com]
Sent: Wednesday, October 02, 2002 8:45 PM
Subject: IBM MSS Security Vulnerability Alert: WORM: W32/Bugbear@MM
Importance: High

IBM Global Services
Managed Security Services
Security Vulnerability Alert

3 OCT 2002 1:28 GMT MSS-SVA-E01-2002:001.1
==
============
-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

VULNERABILITY SUMMARY

VULNERABILITY: WORM: W32/Bugbear@MM

PLATFORMS: Windows

SOLUTION: Update virus definitions / antivirus software and scan all systems

THREAT: Email mass mailer infects via email OR unprotected file shares.

==

DETAILED INFORMATION

I. Description

W32.Bugbear@mm is a mass-mailing worm. It can also spread through network shares. It has keystroke-logging and backdoor capabilities. The worm also attempts to terminate the processes of various antivirus and firewall programs.

Security Response has seen that because the worm does not properly handle the network resource types, it may flood shared printer resources, which causes them to print garbage or disrupt their normal functionality.

It is written in the Microsoft Visual C++ 6 programming language and is compressed with UPX v0.76.1-1.22.

II. Impact

Large scale e-mailing: Attemps to mass-mail to addresses harvested from a compromised host using it's own SMTP engine

Compromises security settings: May allow unauthorized access to compromised machines. Attempts to terminate processes of various antivirus and firewall programs.

III. Solutions

Standard protective procedures include:

Turn off and remove unneeded services. By default, many operating systems install auxiliary services that are not critical, such as an FTP server, telnet, and a Web server. These services are avenues of attack.
* If they are removed, blended threats have less avenues of attack and you have fewer services to maintain through patch updates.

* If a blended threat exploits one or more network services, disable, or block access to, those services until a patch is applied.

* Always keep your patch levels up-to-date, especially on computers that host public services and are accessible through the firewall, such as HTTP, FTP, mail, and DNS services.

* Enforce a password policy. Complex passwords make it difficult to crack password files on compromised computers. This helps to prevent or limit damage when a computer is compromised.

* Configure your email server to block or remove email that contains file attachments that are commonly used to spread viruses, such as .vbs, .bat, .exe, .pif and .scr files.

* Isolate infected computers quickly to prevent further compromising your organization. Perform a forensic analysis and restore the computers using trusted media.

* Train employees not to open attachments unless they are expecting them. Also, do not execute software that is downloaded from the Internet unless it has been scanned for viruses. Simply visiting a compromised Web site can cause infection if certain browser vulnerabilities are not patched.

Removal

You may be able to use a removal tool from your antiviru software vendor to remove this virus. Otherwise manual removal is possible:

Manual Removal

As an alternative to using the removal tool, you can remove this threat manually.

NOTE: These instructions are for all current and recent Symantec antivirus products, including the Symantec AntiVirus and Norton AntiVirus product lines.

1. Update the virus definitions.
2. Restart the computer in Safe mode.
3. Run a full system scan, and delete all files that are detected as W32.Bugbear@mm.
4. Delete the value added by the worm from the registry key

HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\RunOnce

For details see:

http://www.symantec.com/avcenter/venc/data/w32.bugbear@mm.html

IV. Acknowledgements

Symantec: http://www.symantec.com/avcenter/venc/data/w32.bugbear@mm.html
Trend Micro:
http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_BUGBE
AR
.A
F-Secure: http://www.f-secure.com/v-descs/tanatos.shtml
Sophos: http://www.sophos.com/virusinfo/analyses/w32bugbeara.html
McAfee: http://vil.nai.com/vil/content/v_99728.htm

```
-----BEGIN PGP SIGNATURE-----
Version: PGP Personal Privacy 6.5.3

iQA/AwUBPZuSrcXrSKQHhgFwEQLceACgpv3MXP3gHaxslIftlu8IHXtui7YAoIDv
isyWB0vwR+B+77jom9AyKh0L
=i0OY
-----END PGP SIGNATURE-----
=================================================================
============
```

IBM's Managed Security Services (IBM MSS) is a subscription-based Internet security response service that includes computer security incident response and management, regular electronic verification of your Internet gateway(s), and security vulnerability alerts similar to this one that are tailored to your specific computing environment. IBM's Managed Security Services advisory service is a subscription-based service that provides assistance with virus risk and emergency management. By acting as an extension of your own internal security staff, IBM MSS's team of security experts helps you quickly detect and respond to attacks and exposures to your I/T infrastructre.

As a part of IBM's Business Continuity Recovery Services organization, IBM Managed Security Services is a component of IBM's SecureWay(tm) line of security products and services. From hardware to software to consulting, SecureWay solutions can give you the assurance and expertise you need to protect your valuable business resources. To find out more about IBM Managed Security Services, send an electronic mail message to ers-sales@ers.ibm.com, or call 1-800-426-7378.

IBM MSS maintains a site on the World Wide Web at
http://www-1.ibm.com/services/continuity/recover1.nsf/ers/mss+home
Visit the site for information about the service, copies of security alerts, team contact information, and other items.

IBM MSS uses Pretty Good Privacy* (PGP*) as the digital signature mechanism for security vulnerability alerts and other distributed information. The IBM MSS PGP* public key is available from
http://www-1.ibm.com/services/continuity/recover1.nsf/mss/PGP
"Pretty Good Privacy" and "PGP" are trademarks of Philip Zimmermann.

IBM MSS is a Member Team of the Forum of Incident Response and Security Teams (FIRST), a global organization established to foster cooperation and response coordination among computer security teams worldwide.

Copyright 2002 International Business Machines Corporation.

The information in this document is provided as a service to customers of IBM Managed Security Services. Neither International Business Machines Corporation, nor any of its employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, complete- ness, or usefulness of any information, apparatus, product, or process contained herein, or represents that its use would not infringe any privately owned rights. Reference herein to any specific commercial products, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation or favoring by IBM or its subsidiaries. The views and opinions of authors expressed herein do not necessarily state or reflect those of IBM or its subsidiaries, and may not be used for advertising or product endorsement purposes.

The material in this security alert may be reproduced and distributed, without permission, in whole or in part, by other security incident response teams (both commercial and non-commercial), provided the above copyright is kept intact and due credit is given to IBM MSS.

This security alert may be reproduced and distributed, without permission, in its entirety only, by any person provided such reproduction and/or distribution is performed for non-commercial purposes and with the intent of increasing the awareness of the Internet community.
===
============

Appendix C

MIME Media Types[19]

> [RFC2045,RFC2046] specifies that Content Types, Content Subtypes, Character Sets, Access Types, and conversion values for MIME mail will be assigned and listed by the IANA.

The following is the list of Directories of Content Types.

MIME content-types supported by most web servers and identified with file extensions, are listed in the following table.[20]

MIME Type	Identification	File Extension
application/acad	AutoCAD	dwg
application/arj	compressed archive	arj
application/astound	Astound	asd, asn
application/clariscad	ClarisCAD	ccad
application/drafting	MATRA Prelude drafting	drw
application/dxf	DXF (AutoCAD)	dxf
application/i-deas	SDRC I-DEAS	unv
application/iges	IGES graphics format	iges, igs
application/java-archive	Java archive	jar
application/mac-binhex40	Macintosh binary BinHex 4.0	hqx
application/msaccess	Microsoft Access	mdb
application/msexcel	Microsoft Excel	xla, xls, xlt, xlw
application/mspowerpoint	Microsoft PowerPoint	pot, pps, ppt
application/msproject	Microsoft Project	mpp
application/msword	Microsoft Word	doc, word, w6w
application/mswrite	Microsoft Write	wri
application/octet-stream	uninterpreted binary	bin
application/oda	ODA	oda
application/pdf	Adobe Acrobat	pdf
application/postscript	PostScript	ai, eps, ps
application/pro_eng	PTC Pro/ENGINEER	part, prt
application/rtf	Rich Text Format	rtf
application/set	SET (French CAD)	set

[19] http://perl.about.com/gi/dynamic/offsite.htm?site=http://www.iana.org/assignments/media%2Dtypes/index.html

[20] http://www.hansenmedia.com/mime_typ.htm

application/sla	stereolithography	stl
application/solids	MATRA Prelude Solids	sol
application/STEP	ISO-10303 STEP data	st, step, stp
application/vda	VDA-FS Surface data	vda
application/x-bcpio	binary CPIO	bcpio
application/x-cpio	POSIX CPIO	cpio
application/x-csh	C-shell script	csh
application/x-director	Macromedia Director	dcr, dir, dxr
application/x-dvi	TeX DVI	dvi
application/x-dwf	AutoCAD	dwf
application/x-gtar	GNU tar	gtar
application/x-gzip	GNU ZIP	gz, gzip
application/x-hdf	NCSA HDF Data File	hdf
application/x-javascript	JavaScript	js
application/x-latex	LaTeX source	latex
application/x-macbinary	Macintosh compressed	bin
application/x-midi	MIDI	mid
application/x-mif	FrameMaker MIF	mif
application/x-netcdf	Unidata netCDF	cdf, nc
application/x-sh	Bourne shell script	sh
application/x-shar	shell archive	shar
application/x-shockwave-flash	Macromedia Shockwave	swf
application/x-stuffit	StuffIt archive	sit
application/x-sv4cpio	SVR4 CPIO	sv4cpio
application/x-sv4crc	SVR4 CPIO with CRC	sv4crc
application/x-tar	4.3BSD tar format	tar
application/x-tcl	TCL script	tcl
application/x-tex	TeX source	tex
application/x-texinfo	Texinfo (Emacs)	texi, texinfo
application/x-troff	Troff	roff, t, tr
application/x-troff-man	Troff with MAN macros	man
application/x-troff-me	Troff with ME macros	me
application/x-troff-ms	Troff with MS macros	ms
application/x-ustar	POSIX tar format	ustar
application/x-wais-source	WAIS source	src
application/x-winhelp	Microsoft Windows help	hlp
application/zip	ZIP archive	zip
audio/basic	BASIC audio (u-law)	au, snd

audio/midi	MIDI	mid, midi
audio/x-aiff	AIFF audio	aif, aifc, aiff
audio/x-mpeg	MPEG audio	mp3
audio/x-pn-realaudio	RealAudio	ra, ram
audio/x-pn-realaudio-plugin	RealAudio plug-in	rpm
audio/x-voice	Voice	voc
audio/x-wav	Microsoft Windows WAVE audio	wav
image/bmp	Bitmap	bmp
image/gif	GIF image	gif
image/ief	Image Exchange Format	ief
image/jpeg	JPEG image	jpe, jpeg, jpg
image/pict	Macintosh PICT	pict
image/png	Portable Network Graphic	png
image/tiff	TIFF image	tif, tiff
image/x-cmu-raster	CMU raster	ras
image/x-portable-anymap	PBM Anymap format	pnm
image/x-portable-bitmap	PBM Bitmap format	pbm
image/x-portable-graymap	PBM Graymap format	pgm
image/x-portable-pixmap	PBM Pixmap format	ppm
image/x-rgb	RGB image	rgb
image/x-xbitmap	X Bitmap	xbm
image/x-xpixmap	X Pixmap	xpm
image/x-xwindowdump	X Window System dump	xwd
multipart/x-gzip	GNU ZIP archive	gzip
multipart/x-zip	PKZIP archive	zip
text/html	HTML	htm, html
text/plain	plain text	C, cc, h, txt
text/richtext	MIME Richtext	rtx
text/tab-separated-values	text with tabs	tsv
text/x-setext	Structurally Enhanced Text	etx
text/x-sgml	SGML	sgm, sgml
video/mpeg	MPEG video	mpe, mpeg, mpg
video/msvideo	Microsoft Windows video	avi
video/quicktime	QuickTime video	mov, qt
video/vdo	VDO streaming video	vdo
video/vivo	VIVO streaming video	viv, vivo
video/x-sgi-movie	SGI Movieplayer format	movie
x-conference/x-cooltalk	CoolTalk	ice

x-world/x-svr	Virtual reality	svr
x-world/x-vrml	VRML Worlds	wrl
x-world/x-vrt	Virtual reality	vrt