Global Information Assurance Certification Paper

GIAC Certified Incident Handler (GCIH)
Practical Assignment v3

Swen: The Worm with Social Engineering Aspirations

Reid Stephan, CISSP, MCSE, CCNA
December 16, 2003

TABLE OF CONTENTS

1. STATEMENT OF PURPOSE

The purpose of this paper is to describe a fictional, yet possible Information Technology (IT) security incident. The paper will address the details of the exploit, the platforms and environments involved, the stages of the attack (including the business impact), and provide a detailed description and evaluation of the incident handling process in response to the incident.

The attack will be written from the point of view of a fictitious character named Brady Paulson. Brady's brother, Eric, had worked for Example.com for over twenty years but was recently laid off without so much as a thank you. Eric had taken this very hard and was suffering from a serious case of depression. Outraged at they way his brother had been treated, Brady vowed to get revenge.

A computer enthusiast, Brady described himself as a self-taught hacker. He kept current on vulnerabilities and exploits through sites such as http://packetstormsecurity.nl/ and various newsgroups in the hacker community.

Recently, a worm called Swen was beginning to spread and cause mayhem. Brady had acquired a copy of the infected HTML e-mail message, and decided that would be just the medicine to give Example.com. Swen would cause disruption to Example.com's normal business processes in the form of bandwidth and resource consumption. Brady's aim is to cause disruption and chaos that will hopefully lead to financial loss and public embarrassment for Example.com. Infection will be achieved by e-mailing the worm to a broad base of Example.com employees. Due to the various means of propagation and the speed associated with it, the attack can be successful even if only a small percentage of users are initially infected.

2. THE EXPLOIT

In terms of malicious computer code, a worm is a self-contained program (or set of programs) that is able to spread working copies of itself to other computer systems. The most common means of distribution is through e-mail and network connections [1].

NAME

Swen is a worm that became publicly prevalent on September 18, 2003. Various anti-virus vendors and researchers christened it with different titles, which are listed in *Table 1.*

Vendor	Name
CA	Win32 Swen.A
F-Secure	Swen
KAV	I-Worm.Swen
McAfee	W32/Swen@mm
Sophos	W32/Gibe-F
Symantec	W32/Swen.A@mm
Trend Micro	WORM_SWEN.A

Table 1: W32/Swen names

For the purposes of this paper, this version of the Swen worm will be referenced as W32/Swen.

W32/Swen poses a particular problem in that it is a blended threat, meaning that it is able to spread via a variety of methods. These methods include e-mail, file-sharing networks, newsgroups, network shares, and even an exploit of a Microsoft Internet Explorer vulnerability. It is also high polymorphic, with a good deal of intelligence built into it. This allows it to use different subject lines, attachment names, and message bodies in the e-mail it spawns, making it increasingly difficult to detect and stop. Moreover, it can also present an HTML e-mail that masquerades as a security update from Microsoft. While the e-mail is not perfect, it is a clever enough social engineering twist that fools many end users.

As stated, W32/Swen can exploit a known vulnerability in Internet Explorer 5.01 and 5.5 that causes unusual MIME types to be processed incorrectly in HTML e-mail. This will be discussed in greater detail later in this paper. CVE-2001-0154 covers this vulnerability:

http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2001-0154

OPERATING SYSTEM

W32/Swen can affect the following systems, irrespective of service pack levels or patching status:

- Windows NT
- Windows 95
- Windows 98
- Windows ME
- Windows 2000
- Windows XP
- Windows 2003 Server

PROTOCOLS / SERVICES / APPLICATIONS

W32/Swen can propagate through a variety of methods, including:

E-mail

- Any e-mail client can receive a copy of the infected message.
- Outlook, Outlook Express, and Web-based mail clients are susceptible to the IE vulnerability.
- E-mail is sent using the Simple Mail Transport Protocol (SMTP), which allows for the efficient and reliable transfer of e-mail. SMTP is primarily transported over the Transmission Control Protocol (TCP), although other transports are possible. A key component of SMTP is its ability to send mail across networks (known as mail relaying). For our purpose, networks can be defined as available TCP hosts on the public Intranet or a private Intranet.

  An SMTP client that wishes to send e-mail will locate an SMTP server to pass the message off to. If the receiving SMTP server is the ultimate destination, then the message will delivered to the intended recipient. If the receiving SMTP server is not the ultimate destination, then it becomes an intermediate relay (in essence an SMTP client) and will deliver the message to the next appropriate

SMTP server. What is important to note here is that a message can be delivered in single connection between the sender and receiver, or the delivery can occur via a series of hops through relay servers. An improperly configured SMTP server can be exploited to relay mail messages that it should not. This can be taken advantage of by parties who wish to send messages that falsify the source address or route.

Here is the list of SMTP commands:

HELO
(Replaced by EHLO in new versions)
Used to greet the mail server. This is used at the beginning of each session.

MAIL FROM: whoisitfrom
This identifies the sender. This is used once per message, prior to specifying any recipients or after the RSET command is issued.

RCPT TO: whoisitfor
This identifies who the recipient is. Multiple recipients can be entered, as long as each has its own RCPT TO: entered.

DATA
This begins the mail entry mode. Everything entered on the lines following DATA is treated as the body of the message and is sent to the recipients. The DATA session ends with a . (period) on a line by itself. One the period is entered, the mail may be queued or sent immediately.

RSET
This resets the state of the current transaction. Any entries to the MAIL FROM: and RCPT TO: for the current transaction are cleared. This cannot be used once the DATA command has been issued.

QUIT
This ends the session.

- Ports
    - tcp/25 Simple Mail Transport Protocol (SMTP) [2]

Internet Relay Chat (IRC)

- Any user connected to an IRC channel with an W32/Swen infected user is susceptible.

- IRC [3] provides a method of real-time chat for users worldwide. It consists of a publicly available network of IRC servers that allow users to connect using an IRC client. There are several freely available clients for users to choose from, such as ircII (*nix) and mIRC (Windows). Users then join “channels” (virtual locations, usually based on topic of conversation) to chat privately, in groups, or share files.

- Ports used:

    - tcp/6667 & udp/6667 usually

    - tcp & udp/6660-6670 is the most common range

Peer-to-Peer (P2P) file sharing networks

- P2P networks are decentralized systems where each computer has similar capabilities and responsibilities and can act as both the client and the server. Users of a P2P network are able to share files with each other as they see fit. To participate in a P2P network, users must install the necessary client software. There are several freely available P2P software clients available, such as KaZaA and Gnutella.

- P2P protocols are mainly proprietary. Gnutella uses the open source gnutella protocol [4], while KaZaA use the Fast Track protocol [5].

- Ports used:

    - Gnutella uses tcp/6346 by default

    - KaZaA uses tcp/1214 by default

    - P2P clients can be configured to use different ports

Newsgroups

- W32/Swen attempts to use enticing file names (social engineering) to get users to download it.

- Newsgroups provide a way for users to communicate about topics in an asynchronous, written method. The conversations are stored in a central database and offer the subscriber the ability to only select the items they wish to read. Newsgroups function using the Network News Transfer Protocol (NNTP) [6].

- Ports used:

    - tcp/119 and tcp/563 (Secure Socket Layer)

Network Shares

- W32/Swen attempts to use enticing file names (social engineering) to get users to download it.

- Network shares allow files and folders on a system to be shared and accessed by other systems on the network.

- Ports used:

    - udp/137 and tcp/139

VARIANTS

There were several predecessors to W32/Swen. These included W32/Gibe.A, W32/Gibe.B, W32/Gibe.C, and W32/Gibe.D. There similarities and differences to W32/Swen are listed in *Table 2*.

W32/Swen Predecessors	Similarities to W32/Swen	Differences from W32/Swen
W32/Gibe.B	• Uses its own SMTP engine • Spread though network shares • Disguised as a security update • Affects the same operating systems	• Written in Visual Basic where as W32/Swen is written in Microsoft Visual C/C++ (MSVC) • Attachment always named Q216309.exe (W32/Swen uses random file names) • Installs a backdoor Trojan
W32/Gibe.B	• Uses its own SMTP engine • Spreads though P2P • Spreads through IRC • Spreads through network shares • Spreads through newsgroups • Disguised as a security update • Exploits MS01-020 • Uses multiple attachment names	• Written in Visual Basic where as W32/Swen is written in MSVC • Copies itself as Webloader.exe to the startup folder of all mapped drives.

W32/Swen Predecessors	Similarities to W32/Swen	Differences from W32/Swen
W32/Gibe.B (con)	• Affects the same operating systems • Uses its own SMTP engine • Spreads though P2P • Spreads through IRC • Disguised as a security update • Exploits MS01-020	• Written in Visual Basic where as W32/Swen is written in MSVC
W32/Gibe.D	• Uses its own SMTP engine • Spreads though P2P • Spreads through IRC • Disguised as a security update • Exploits MS01-020	• Written in Visual Basic where as W32/Swen is written in MSVC

Table 2: W32/Swen Predecessor Similarities & Difference

Two additional variants of this worm have been observed in the wild. The variants behave in the same manner as W32/Swem. The variance is created by minor edits of certain strings within the initial worm, and subsequent packing with UPX (a utility used to package executables). They are detected by the same virus definition files that detect W32/Swen. On notable difference is that the file size of the variants is 52,224 bytes where W32/Swen is 106,496 bytes long.

DESCRIPTION & SIGNATURES OF THE ATTACK

W32/Swen is a blended threat worm that propagates via e-mail, IRC channels, network shares, newsgroups, and P2P file sharing networks. A system can become infected through running the infected executable, or by simply viewing or opening an infected e-mail with an e-mail client that relies on a vulnerable version of Internet Explorer to read HTML messages. The infected executable is 106,496 bytes in length [7].

When the W32/Swen worm is executed, it performs several actions:

1) It will first determine if it is already installed on the target system. If a previous installation of W32/Swen is detected, it will abort the installation and display the figure shown in *Figure 1*.

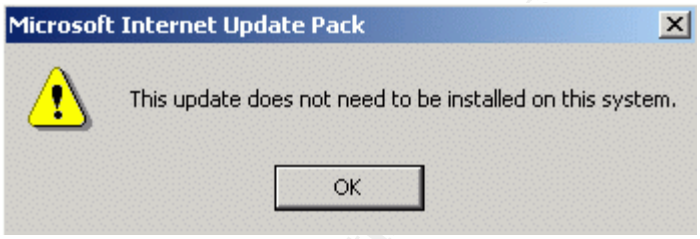

Figure 1: Image shown by W32/Swen if already installed

2) If it determines W32/Swen is not already installed, it will continue the installation and present the user with the dialog box shown in *Figure 2*.

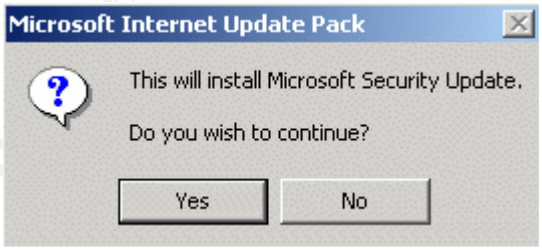

Figure 2: Image shown by W32/Swen if not installed

3) Regardless of what choice the user makes, the W32/Swen will be installed at this point. If the user clicks “No”, then the installation will take place silently. If the user clicks “Yes”, then the dialog box shown in *Figures 3 and 4* will be shown.

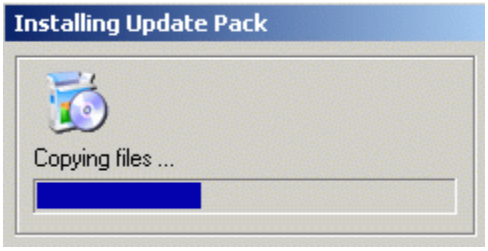

Figure 3: Image shown by W32/Swen during installation

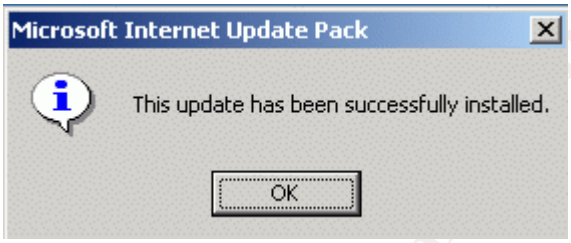

Figure 4: Image shown by W32/Swen during installation

4) Once installed, W32/Swen attempts to stop processes that represent anti-virus programs, personal firewall programs, and other utilities that may interfere with the worm's functionality. A list of these impacted processes is shown in *Table 3* [8].

Table 3: List of processes W32/Swen attempts to stop

_avp	dv95	kpfw32	pavw
Azonealarm	espwatch	luall	pavsched
avwupd32	esafe	lookout	pavcl
avwin95	efinet32	lockdown2000	padmin
avsched32	ecengine	msconfig	rescue
avp	f-stopw	mpftray	regedit
avnt	frw	mpftray	rav
avkserv	fp-win	moolive	sweep
avgw	f-prot95	nvc95	sphinx
avgctrl	fprot95	nupgrade	serv95
avgcc32	f-prot	nupdate	safeweb
ave32 7	fprot	normist	tds2
avconsol	findviru	nmain	tca
autodown	f-agnt95	nisum	vsstat
apvxdwin	gibe	navw	vshwin32
aplica32	iomon98	navsched	vsecomr
anti-trojan	iface	navnt	vscan
ackwin32	icsupp	navlu32	vettray
bootwarn	icssuppnt	navapw32	vet98
blackice	icmoon	nai_vs_stat	vet95
blackd	icmon	outpost	vet32

claw95 cfinet cfind cfiaudit cfiadmin ccshtdwn ccapp	icloadnt icload95 ibmavsp ibmasn iamserv iamapp jedi	pview pop3trap persfw pcfwallicon pccwin98 pccmain pcciomon	vcontrol vcleaner wfindv32 webtrap zapro

Table 3: List of processes W32/Swen attempts to stop (con)

If W32/Swen intercepts the execution of any of these processes, it will stop them from loading and display the dialog box shown in *Figure 5*:

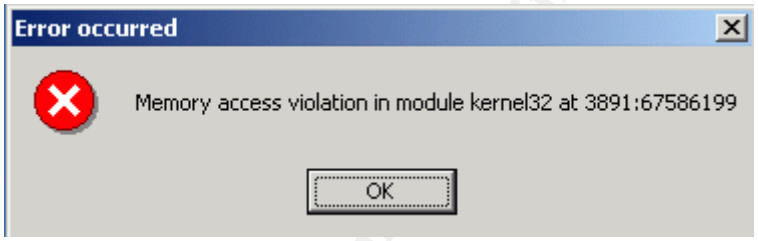

Figure 5: Image shown by W32/Swen when execution of a stopped process is attempted

5) W32/Swen will then make a copy of itself in %Windir% (%Windir% represents the Windows installation folder). By default this is C:\Windows or C:\Winnt. W32/Swen will actually query the operating system to determine the exact location of the Windows installation folder. It will use random letters for the filename.

6) It also copies a %ComputerName%.bat (%ComputerName% represents the machine name of the infected computer) file, which launches the worm and randomly names a configuration file that is used to store data specific to the local machine. For example, if the worm were named kmbv.exe, the contents of the configuration file would be:

```
@ECHO OFF
IF NOT "%1"=="" kmbv.exe %1
```

7) W32/Swen will then harvest email addresses from the computer by searching all files with the following extensions [7]:

- .asp (Active Server Page files)
- .dbx (Outlook Express email folder)
- .eml (Used by e-mail clients, including Outlook Express)
- .ht* (HTML files)
- .mbx (Mailbox messaging file, used by Outlook v1-4, Eudora & others)
- .wab (Outlook address book)

8) W32/Swen then creates a file named %Windir\Germs0.dbv to store all the harvested e-mail addresses.

9) W32/Swen also creates a file named %Windir\Swen1.dat to store a list of remote mail and news servers. See *Appendix A* for a list of these servers.

10) The following values [7]:

- "CacheBox Outfit"="yes"
- "ZipName"="<random>"
- "Email Address"="<The current users email address that the worm retrieves from the registry>"
- "Server"="<The IP address of the SMTP server that the worm retrieves from the registry>"
- "Mirc Install Folder"="<location of mirc client on system>"
- "Installed"="...by Begbie"
- "Install Item"="<random>"
- "Unfile"="<random>"

are added to the registry[1] key:

- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Current Version\explorer* (* represents a random set of letters)

11) The random name assigned to the worm is added to the registry key [7]:

- HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Current Version\Run

This enables the worm to start each time Windows starts.

12) The following registry keys:

- HKEY_LOCAL_MACHINE\Software\CLASSES\exefile\shell\open\ command
- HKEY_LOCAL_MACHINE\Software\CLASSES\regfile\shell\open\ command
- HKEY_LOCAL_MACHINE\Software\CLASSES\scrfile\shell\open\ command
- HKEY_LOCAL_MACHINE\Software\CLASSES\comfile\shell\open\ command
- HKEY_LOCAL_MACHINE\Software\CLASSES\batfile\shell\open\ command
- HKEY_LOCAL_MACHINE\Software\CLASSES\piffile\shell\open\ command

Are modified so that the worm will be run prior to .EXE, .SCR, .COM, .BAT, and .PIF files being executed. It will also display a bogus error message whenever.REG files are opened (see *Figure 5*) [9].

---

1 The *Microsoft Computer Dictionary*, Fifth Edition, defines the registry as:

> "A central hierarchical database used in Microsoft Windows 9x, Windows CE, Windows NT, and Windows 2000 used to store information necessary to configure the system for one or more users, applications and hardware devices."

13) Occasionally, W32/Swen will present the user with a fake MAPI32 Exception error, as shown in *Figure 6*:

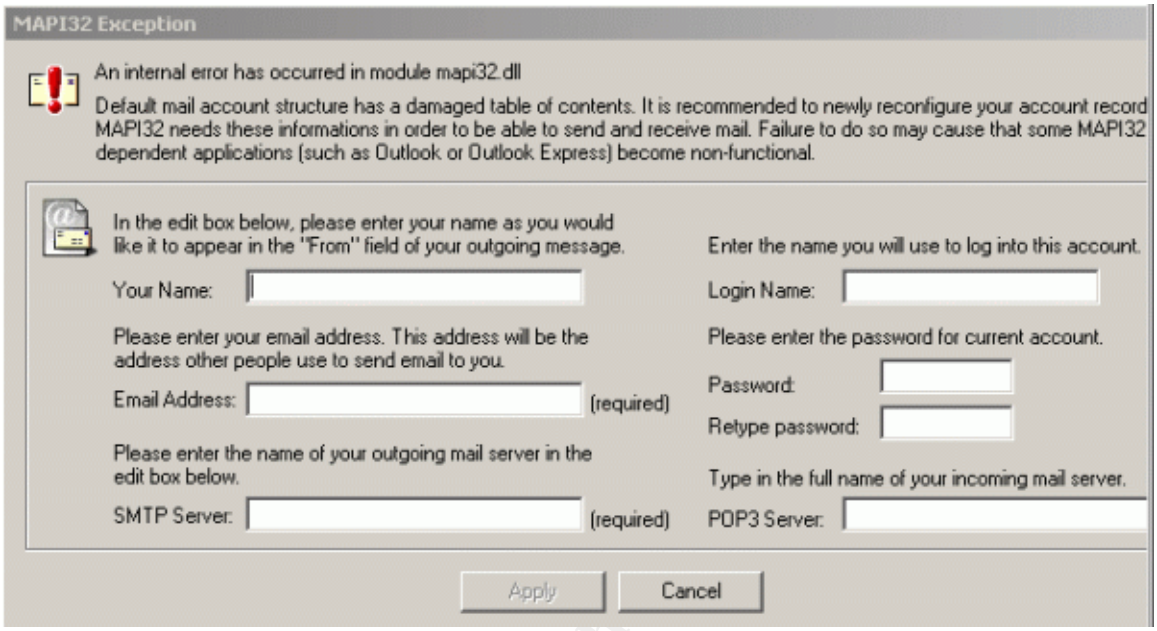

Figure 6: Image of the W32/Swen bogus MAPI32 error

Assuming a user entered the requested information, W32/Swen would then use this information to log into the POP3 mail server and delete any email that the worm sent from the infected machine.

14) Several copies of the worm may exist on the computer. W32/Swen keeps track by listing all copies of itself in a file in the Windows directory called esueif.vcn [11].

15) As mentioned previously, W32/Swen attempts to propagate through email, P2P, IRC, mapped drives, and newsgroups.

HOW W32/SWEN SPREADS VIA E-MAIL

W32/Swen will send a copy of itself to any e-mail addresses it was able to harvest from the infected machine. The worm is particularly nefarious in that it can vary the wording of the subject line, the message it sends, and the filename it attaches itself as.

In terms of an e-mail subject line, W32/Swen will embody one of two formats [10].

Format 1

In this variation, the e-mail subject line will contain up to fours strings, and may also be lowercase:

String 1	String 2	String 3	String 4
<Empty>	<Empty>	<Empty>	Pack
Current	Internet	Critical	Patch
Last	Microsoft	Secure	Update
Latest	Net		Upgrade
New	Network		
Newest			

Format 2

In this variation, the e-mail subject line will contain up to 13 strings:

String 1	String 2	String 3
<Empty>	<Empty>	<Empty>
FW:	Apply	that
FWD:	Check	this
RE:	Checkout	the
	Install	these
	LookAt	
	Prove	
	TakALookAt	
	Taste	
	Try	
	TryOn	
	See	
	Use	
	Watch	

String 4	String 5	String 6
<Empty>	pack	<Empty>
correction	package	for
corrective	patch	
critical	<The subject line for	
important	Format 2 may end here>	
internet		
security		

String 7	String 8	String 9
<Empty if String 6 is empty>	<Empty>	<Empty if String 8 is empty>
Internet Explorer	that	came
Windows	which	comes
<The subject line for Format 2 may end here>		

String 10	String 11	String 12
from	<Empty>	Microsoft
	the	MS
		M$

String 13
<Empty>
Corp.
Corporation

The attachment name is created by taking one of the following static names:

- Patch
- Upgrade
- Update
- Installer
- Install
- Pack
- Q

and adding a series of random numbers and an .exe or .zip file extension [7].

The body of the email can attempt to appear is if it is a security alert notice from Microsoft Corporation, as shown in *Figure 7*.

Figure 7: Example of HTML e-mail generated by W32/Swen

To become infected with W32/Swen, the user would need to launch the attachment that comes with the message. Alternatively, a user who has not applied the MS01-020 patch, can become infected by simply opening the message or viewing it in the Outlook preview pane.

The Microsoft Security Bulletin MS01-020 was originally posted on March 29, 2001. It affects the following Microsoft products:

- Microsoft Internet Explorer 5.01 (SP1 and below)
- Microsoft Internet Explorer 5.5

The security bulletin disclosed that there is a flaw in the way Internet Explorer processes certain unusual MIME types. MIME is an acronym that stands for Multipurpose Internet Mail Extension. It is a commonly used standard for encoding binary files as e-mail attachments. E-mail that contains a binary attachment must specify the file type so that the receiving e-mail program can correctly interpret it. There are certain unusual MIME types that Internet Explorer does not handle correctly. Instead of prompting the user with a warning, Internet Explorer will simply execute the unusual MIME attachment. To exploit this vulnerability, an attacker would need to have an unpatched computer browse to a website or view an HTML e-mail that contained a malicious executable disguised as an unusual MIME attachment [12].

So in the case of W32/Swen, the worm pretends to be one of these unusual MIME types. When HTML e-mail is delivered to a computer that has not applied the MS01-020 patch, the worm is launched when the e-mail is opened up, or even viewed in the Outlook preview pane.

The text of the body can also attempt to disguise itself as a mail delivery failure noticewhile attaching itself as a randomly named executable. The wording of such a message would be similar to the following [13]:

```
I'm sorry I wasn't able to deliver your message to one
or more destinations.
```

HOW W32/SWEN SPREADS VIA IRC

To spread via IRC, W32/Swen searches the infected computer for a \Mirc folder. If it finds the \Mirc folder, it attempts to locate the Script.ini file. If the Script.ini file is found, it is renamed to Script.bcp. The new Script.ini file is about 123 bytes in size, and will cause W32/Swen to be run in memory with IRC, uploading copies of itself to other users on the same IRC channel. The W32/Swen worm executable is compressed, and renamed a random file name before being sent to other IRC users. The random naming scheme occurs using one of the string combinations listed below [8]:

String 1 (Used by itself)
Virus Generator
Magic Mushrooms Growing
Cooking with Cannabis

Hallucinogenic Screensaver
My naked sister
XXX Pictures Sick Joke
XXX Video
XP update
Emulator PS2
XboX Emulator
HardPorn
Jenna Jameson
10.000 Serials
Hotmail hacker
Yahoo hacker
AOL hacker

String 2	combined with	String 3
Sobig		fixtool
Sircam		cleaner
Bugbear		removal tool
		Remover

String 4	combined with	String 5
Windows Media Player		installer
GerRight FTP		upload
Download Accelerator		warez
		hacked
		Kkey generator

HOW W32/SWEN SPREADS VIA P2P NETWORKS

To spread through the KaZaAa P2P network, W32/Swen uses enticing file names to trick users into downloading. A P2P user would need to download and execute a W32/Swen infected file in order to become infected. On an infected machine, W32/Swen places a .zip or .rar copy of itself in the %Temp% subdirectory on the infected computer. In this case, %Temp% is a variable. W32/Swen will locate the Windows installation folder (C:Windows or C:Winnt by default) and place a copy there.

In the registry key:

- HKEY_CURRENT_USER\Software\Kazaa\LocalContent

The following values are added:

- "Dir99"= "012345:<random folder name>"
- "DisableSharing"="0"

This adds the folder <random folder name> to the list of KaZaAa shared folders on the infected computer. In this way, W32/Swen continues to offers itself to other users on the P2P network.

The infected files shared on the P2P network will always have an .EXE or .ZIP extension, and have randomized filename generated from the following three lists of string variables:

String 1	String 2	String 3
Download Accelerator	Bugbear	AOL hacker
GetRight FTP	cleaner	Cooking with Cannabis
hack	fixtool	Emulator PS2
hacked	Gibe	Hallucinogenic Screensaver
installer	Klez	HardPorn
Kazaa Lite	remover	Hotmail hacker
KaZaA media desktop	remover tool	Jenna Jameson
KaZaA	Sircam	Magic Mushrooms Growing
key generator	Sobig	My naked sister
Mirc	Yaha	Sex
upload		Sick Joke
Winamp		Virus Generator
Windows Media Player		XboX Emulator
WinRar		XP update
WinZip		XXX Pictures
warez		XXX Video
		Yahoo hacker
		10.000 Serials

HOW W32/SWEN SPREADS VIA NEWSGROUPS

W32/Swen will harvest newsgroup server addresses from the registry of the infected machine, and will attempt to post infected messages to the newsgroups. If no newsgroup servers can be found on the infected machine, W32/Swen will randomly select one from the predefined list delivered with the worm (see Appendix A). The infected message will be created following the same logic the worm uses to create infected e-mail messages attempting to entice users to download by using tempting files names (see section on *How W32/Swen Spread via E-mail)*. To get infected, a user would have to download the W32/Swen malicious file and execute it.

How W32/Swen Spreads Via Network Shares

W32/Swen search drives A-Z, attempting to spread via mapped network drives by copying itself to the following locations when available [13]:

- \Win98\Start menu\Programs\Startup
- \Win95\Start menu\Programs\Startup
- \WinMe\Start menu\Programs\Startup
- \Windows\Start menu\Programs\Startup
- \Documents and Settings\All Users\Start menu\Programs\Startup
- \Documents and Settings\Administrator\Start menu\Programs\Startup
- \Documents and Settings\Default User\Start menu\Programs\Startup
- \Winnt\Profiles\All Users\Start menu\Programs\Startup
- \Winnt\Profiles\Administrator\Start menu\Programs\Startup
- \Winnt\Profiles\Default User\Start menu\Programs\Startup

If successful, the targeted computer will be infected the next time the computer is restarted or a user logs on.

Signatures of the Attack

Besides a manual check for the file and registry entries mentioned above, W32/Swen can also be detected by virus definition files from any major anti-virus software vendor dated 9/18/03 or later.

Also, W32/Swen attempts to track the number of infections by having infected computers visiting a website at:

http://ww2.fce.vutbr.cz/bin/counter.gif/link=bacillus&width=6&set=cnt006

In addition, an MD5[2] hash of the virus payload file could be used by a host file checker to detect the presence of W32/Swen. The payload is always the same executable file that is 106,496 bytes in length. Using the freely available tool m5sum [14], the calculated MD5 hash value of the W32/Swen payload is:

b09e26c292759d654633d3c8ed00d18d

as shown in *Figure 8.* The command used to calculate the MD5 sum of a file is simply *MD5SUM <filename>*.

Figure 8: MD5 hash of W32/Swen payload

This information could be used to manually check for the presence of W32/Swen on a suspected victim machine. Since known behavior of W32/Swen is to copy itself to the %Windir% as a randomly named executable, an MD5 hash performed against all executable files in the %Windir% directory will show if any have the same MD5 hash value that the W32/Swen payload does. *Figure 9* shows the results of such a hash, with the highlighted MD5 value showing that the file fgqweve.exe has the same MD5 value as the patch46.exe W32/Swen payload. In addition, this information could be used by host file checkers to monitor for the presence of W32/Swen.

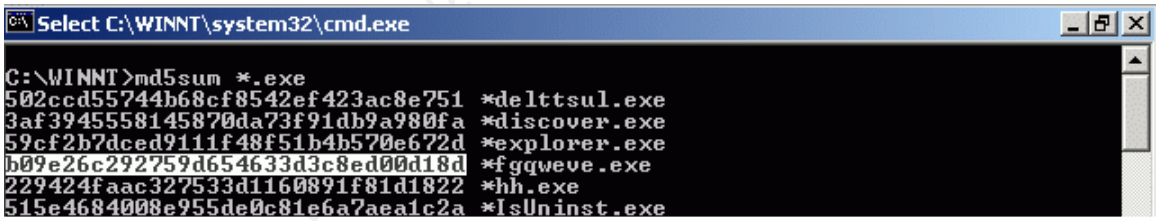

Figure 9: Search results for matching W32/Swen MD5 hash

[2] MD5 stands for Message Digest, extension 5. It was developed by Professor Ronald R. Rivest of MIT, and is an algorithm that can take an input (message or file) and produce a 128-bit message digest of the input. The premise is that it is mathematically infeasible to produce two different messages or files that have the same message digest. MD5 is a way to verify data integrity. Further details can be found in RFC132 (http://www.faqs.org/rfcs/rfc1321.html).

3. THE PLATFORMS/ENVIRONMENTS

VICTIM'S PLATFORM

Hardware: Desktop PC
Pentium III 1 GHz
512 MB RAM
20GB HDD

OS: Windows 2000 Professional, SP4

Applications: Office 2000
Internet Explorer 5.5 SP2
Outlook 2000 (default mail client)
Due to company policy and client build:
- ✓ No P2P software is installed
- ✓ No instant messaging software is installed

AntiVirus: Symantec Corporate Edition, 9/17/03 definition files applied

Patches: All related service packs and security patches are applied

Network Shares: The client is mapped to drive U:\users to store files on a network server that is backed up regularly.

Other than the default admin and RPC shares, no shares exist.

SOURCE NETWORK

The source network is a home network. The home network consists of:

Hardware: Laptop
Pentium III 900MHz
512 MB RAM
10GB HDD

OS: Windows XP Professional, SP1

Applications: Office XP
Internet Explorer 6
Black ICE PC Protection with the following rule base:
- ✓ Block all unsolicited traffic (paranoid)
- ✓ Auto-blocking enabled
- ✓ Internet file-sharing disabled

- ✓ NetBIOS neighborhood not allowed

AntiVirus: Norton AntiVirus 2003, 9/18/03 definition files applied

Patches: All related service packs and security patches have been applied

Connectivity: DSL Internet Connection through local ISP
- ✓ ISP is not known to block any ports
- ✓ MAC address registration is not required

LinkSys EtherFast DSL Router w/ 4-port switch, configured as follows:
- ✓ Connect to ISP via PPPoE (Point-to-Point Protocol over Ethernet)
- ✓ Keep Alive option is set to keep Internet connectivity connected indefinitely
- ✓ No host or domain name are assigned to the router
- ✓ MTU (Maximum Transmission Unit) is set to 1492
- ✓ Local IP address is 192.168.1.1/24
- ✓ Local DHCP server is configured to vend addresses in the 192.168.1.2/24 -192.168.1.10/24 range.
- ✓ Client lease time is set to 1 day.
- ✓ DDNS (Dynamic DNS) is not enabled
- ✓ Public Internet address is obtained dynamically from ISP
- ✓ No static routing is utilized
- ✓ No port range forwarding is utilized
- ✓ No VPN is configured

TARGET NETWORK

Example.com is a leading manufacturer of widgets. Their network consists of a Windows 2000 domain with Windows 2000 clients. They use Exchange 2000 as their e-mail solution. A few remote site locations connect via a dedicated link. The attack does not target a specific portion of the network, but rather has aims to impact the entire user base. All systems connected to the Example.com network have the latest service packs and security patches applied to them. All clients and servers have Norton AntiVirus Corporate Edition installed, and are running virus definition files dated 9/17/03. Example.com sites are connected to the Internet via a T1 and employ a Cisco PIX firewall solution running version 6.1 of the PIX IOS. The firewall rules essentially allow outbound web and mail traffic,

and traffic between Example.com sites. Inbound traffic is allowed to mail servers and web servers. Below is a condensed sample of the common Pix configuration deployed at Example.com.

```
PIX Version 6.1(1)
interface ethernet0 auto
interface ethernet1 auto
nameif ethernet0 outside security0
nameif ethernet1 inside security100
enable password ****
passwd ****
hostname pfw1
domain-name example.com
no fixup protocol smtp 25
access-list inside_out permit tcp any any eq www
access-list inside_out permit udp any any eq domain
access-list inside_out permit tcp any any eq domain
access-list inside_out permit tcp host 10.1.1.128 any
eq smtp
access-list outside_in permit tcp any host
192.0.10.128 eq smtp
access-list outside_in permit tcp any host
192.0.10.130 eq www
ip address outside 192.0.10.123 255.255.255.0
ip address inside 10.1.1.1 255.255.255.0
nat (inside) 1 0.0.0.0 0.0.0.0 0 0
static (inside,outside) 192.0.10.128 10.1.1.128
netmask 255.255.255.255 0 0
static (inside,outside) 192.0.10.130 10.1.1.30 netmask
255.255.255.255 0 0
```

NETWORK DIAGRAM

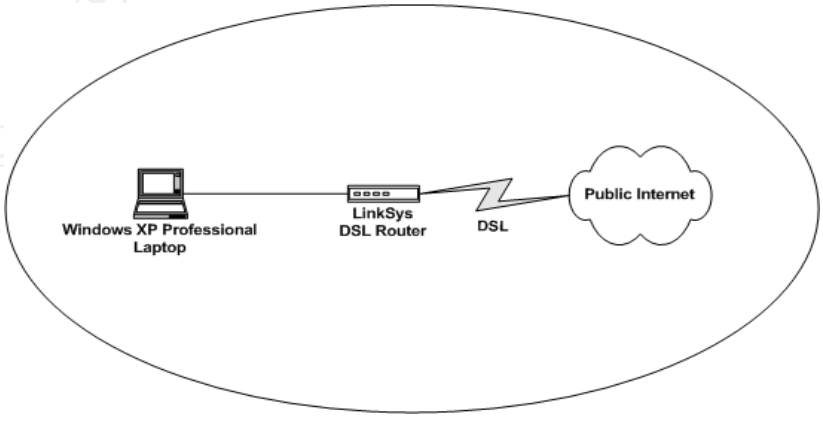

Diagram of Source Network

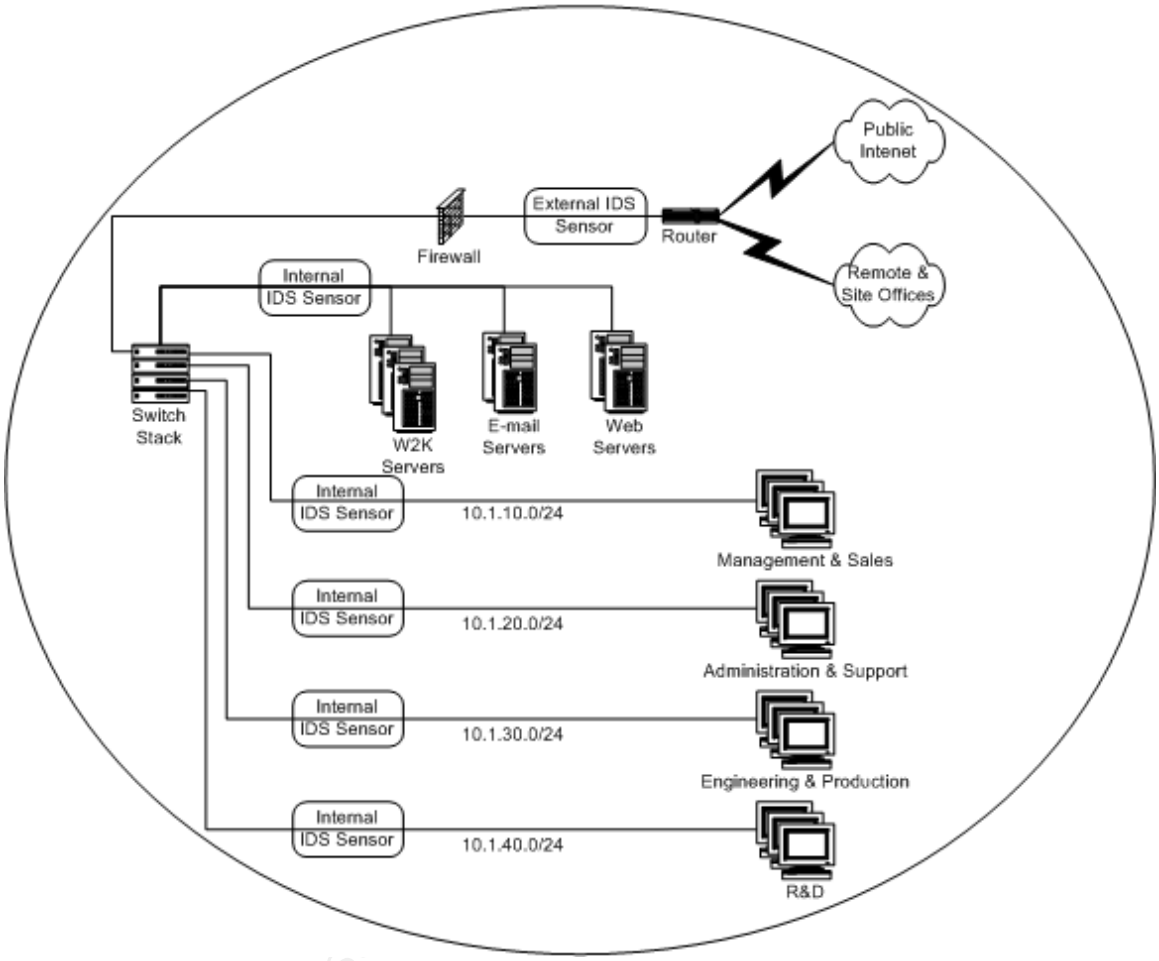

Diagram of Example.com Target Network

4. STAGES OF THE ATTACK

RECONNAISSANCE

From talking to his brother and based on the few times he had visited the office, Brady knew Example.com was a Microsoft shop. Brady's first step was to harvest as many legitimate Example.com e-mail addresses as possible. Since he did not want to involve Eric, he knew he would have to rely on other means. Brady knew the great thing about the Internet is its long memory.

Using the freely available tool SamSpade [15], Brady crawled the www.example.com web space for any available e-mail addresses. The SamSpade web crawler is not the most intelligent one in the world, but it is a good place to start. It will ignore META tags and robots.txt files that are intended to tell crawlers where to look on the website (used for indexing purposes). However, savvy websites that contain self-referential cgi scripts will trap it in an endless recursion. Brady configures the SamSpade crawler to search the .html, .htm, .shtml, .asp, and .txt files of www.example.com and any referenced sites beneath it for e-mail addresses (as shown in *Figure 10*).

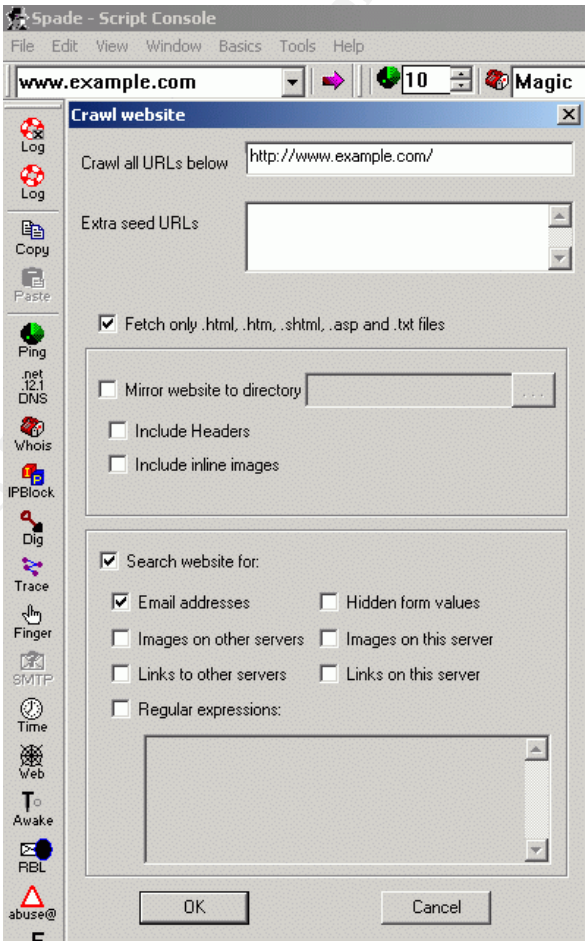

Figure 10: Using Sam Spade to crawl for e-mail addresses

Fortunately for Brady, Example.com was not employing any techniques to prevent or control web crawling. *Figure 11* shows a partial listing of the results Brady received from SamSpade.

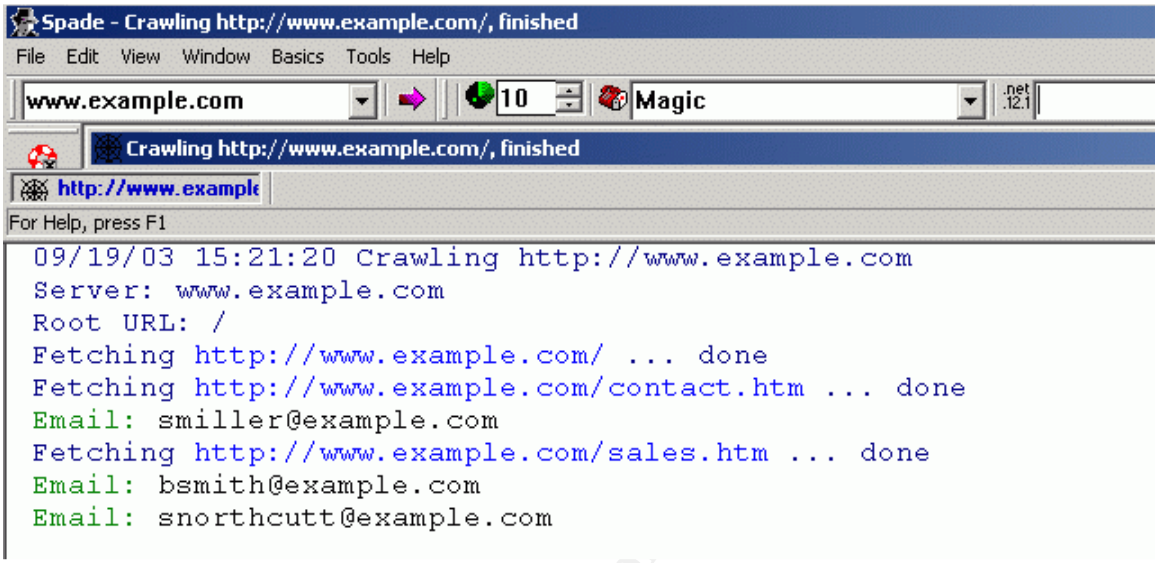

Figure 11: Results from Sam Spade e-mail harvesting

Next, Brady used Google's Advanced Newsgroup search (http://www.google.com/advanced_group_search?hl=en , *Figure 12*) to find any postings that contained Example.com e-mail addresses.

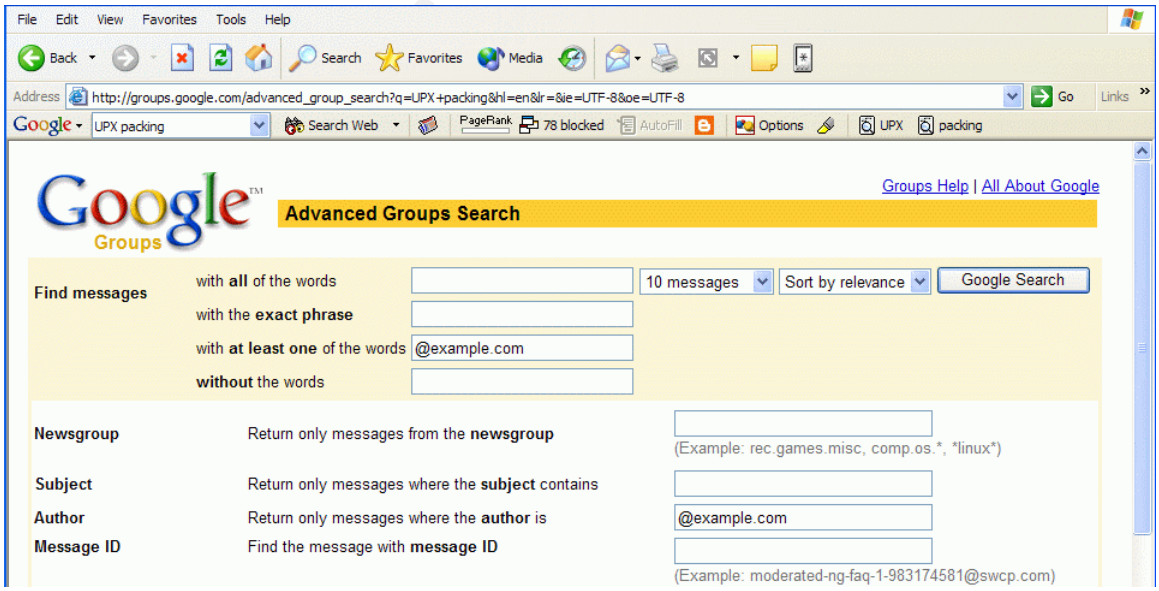

Figure 12: Using Google to harvest e-mail addresses

Using the Google search, there were many techniques Brady employed. First he searched for all messages that had an author of @example.com by entering the seach string "@example.com" in the "*Return all messages where the author is"* field. Then Brady proceeded to search for all messages that had @example.com listed somewhere in the message body by entering the search string "@example.com in the "*Find messages with at least one of the words"* field. Before long, Brady had a substantial list of e-mail addresses to use.

SCANNING

Next, Brady needed a way to send the malicious e-mail in a somewhat anonymous manner. Brady decided he would simply connect to an Example.com mail server and send the infected message to his harvested e-mail addresses using a fake source e-mail address of security@microsoft.com.

First, Brady needed to find the addresses of Example.com mail servers he could use. Using the SamSpade tool, Brady searches for the IP blocks owned by Example.com. This is done by entering www.example.com in the drop down box and then clicking on the IPBlock button on the left hand menu bar. The results are shown in *Figure 13*.

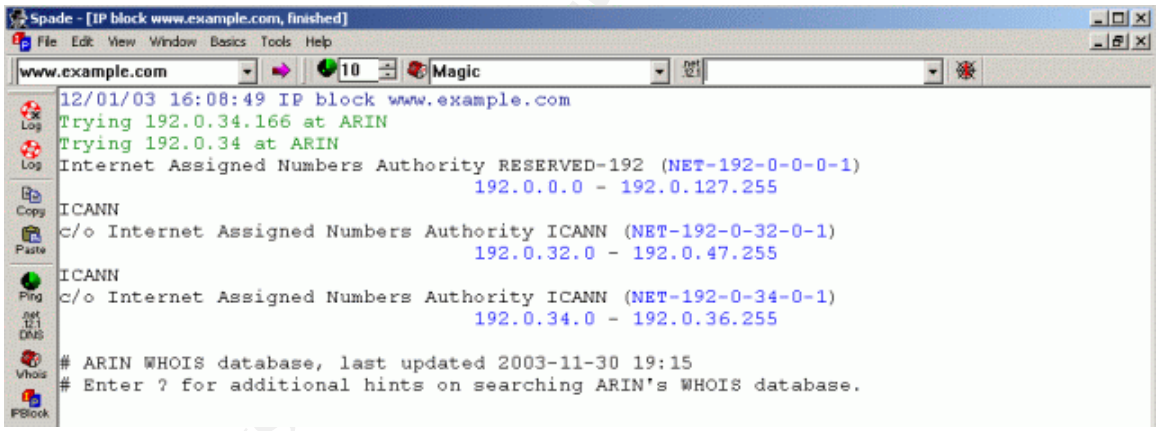

Figure 13: Using Sam Spade to find IP blocks ownership

Brady then uses a nifty utility called SuperScan 3.0 [16] to determine what addresses in the Example.com address range have SMTP port 25 open. As shown in *Figure 14*, SuperScan is configured to scan the address range looking for any host that responds on port 25. The speed is set to minimum to ensure a greater degree of accuracy[3].

[3] Clearly Example.com is a fictitious entity, and is reserved by IANA for testing purposes. As such, the IP address range used here, as well as in previous and future discussions within this document, is for explanation only.

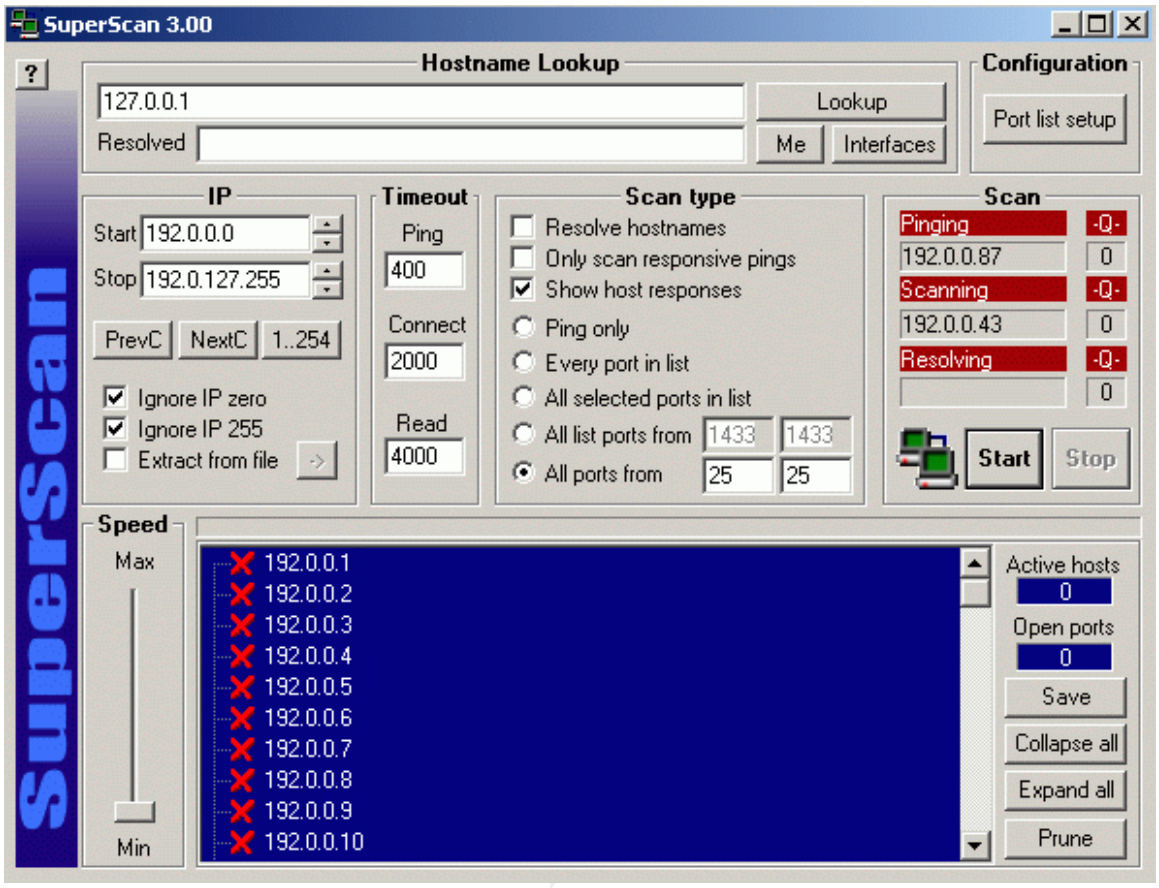

Figure 14: Using SuperScan to detect available SMTP servers

The scan results showed several devices that appeared responsive on port 25. Picking one, Brady opened a command prompt and typed:

```
telnet 192.0.10.128 25
```

–This telnets to the SMTP port (tcp 25) on the targeted system.

The SMTP server responded:

220 Mail01.Example.local SMTP Server Fri, 19 Sep 2003 20:15:55 +0100 (MET)

-Since the SMTP service is listening and responding, it will reply back.

Brady typed:

```
helo Microsoft.com
```

-Brady enters a helo command to introduce himself to the SMTP system.

The SMTP server responded:

250 Mail01.Example.local Hello [192.168.1.2], pleased to meet you

-The SMTP system replies to the greeting. Note that even though Brady had introduced himself as Microsoft.com, the SMTP has the intelligence to determine Brady's actual identity (192.168.1.2).

Brady typed:

```
mail from: bogus.user@microsoft.com
```

-Brady enters the source e-mail address to be used:

The SMTP server responded:

250 bogus.user@microsoft.com... Sender ok

-The SMTP server accepts the source e-mail address:

Brady typed:

```
rcpt to: bogus.user@example.com
```

-Brady enters the destination e-mail address:

The SMTP server responded:

250 2.1.5 bogus.user@example.com

-The SMTP accepts the destination e-mail address, showing that has accepted the source and destination e-mail address.

Brady typed:

```
data
```

-Brady enters the data command, letting the SMTP system know that the message data will follow.

Brady typed:

```
Test.
```

–Brady then enters a test message.

Brady typed:

```
.
```

-When he is done with test message, Brady enters . (period) to let the SMTP system know the message is complete.

The SMTP server responded:

[250 UAA17484 Message accepted for delivery]

-The SMTP system then responds letting Brady know the message is queued for delivery.

Brady typed:

```
quit
```

-Brady exits the telnet session.

Figure 15 shows the input and output from a typical SMTP command session Brady used.

```
C:\>telnet www.example.com 25

Trying 192.0.10.128...
Connected to 192-0-10-128.example.com (192.0.10.128).
Escape character is '^]'.
220 MAIL01.Example.local SMTP Server Fri, 19 Sep 2003 20:15:55 +0100 (MET)
helo microsoft.com
250 MAIL01.Example.local Hello [192.168.1.2]
mail from: bogus.user@microsoft.com
250 2.1.0 bogus.user@microsoft.com....Sender OK
rcpt to: bogus.user@example.com
250 2.1.5 bogus.user@example.com
data
354 Start mail input; end with <CRLF>.<CRLF>
Test
.
250 2.6.0 <MAIL01234xOjIL2jp0000014e@MAIL01.Example.local> Queued mail
for delivery
```

Figure 15: SMTP commands to test for relaying

Brady purposely entered nonexistent source and destination e-mail addresses. The purpose of this test was to simply find a machine that he could use to send the W32/Swen worm. The test message will result in a non-delivery receipt (NDR) message being sent to bogus.user@microsoft.com since bogus.user@example.com is not a legitimate user.

With a usable Example.com mail server in hand[4], Brady is just about ready to send W32/Swen to his gathered list of e-mail recipients.

EXPLOITING THE SYSTEM

Brady then opens Outlook Express and configures a profile with the settings as shown in *Figures 16 and 17*.

[4] In this scenario, Brady found a mail server that allowed mail to be sent to local recipients despite the usage of a fake sender address. Mail servers can be configured to drop mail where the sender's domain does not match up with the resolved source address. If this had been the case, Brady's other options would have been to identify a third party mail server that allows relaying (called an open mail relay) or identify a misconfigured website that allowed mail posting. The point here is that there are several ways in which a forged e-mail can be sent.

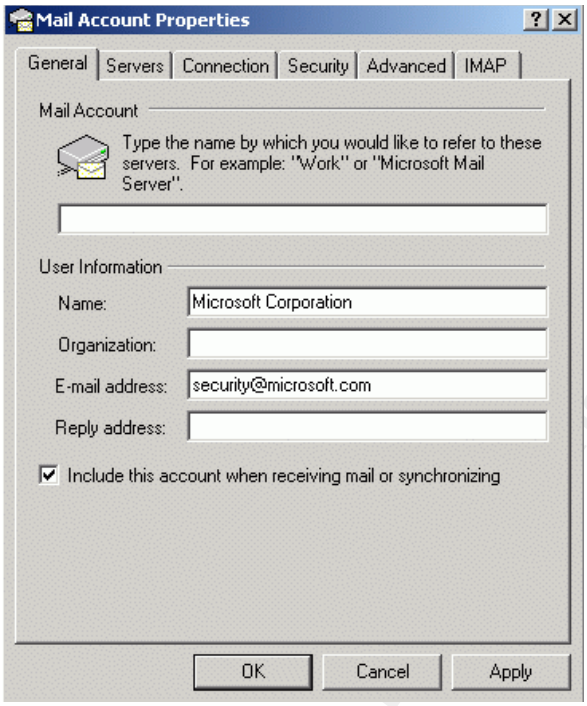

Figure 16: Configuring Outlook Express

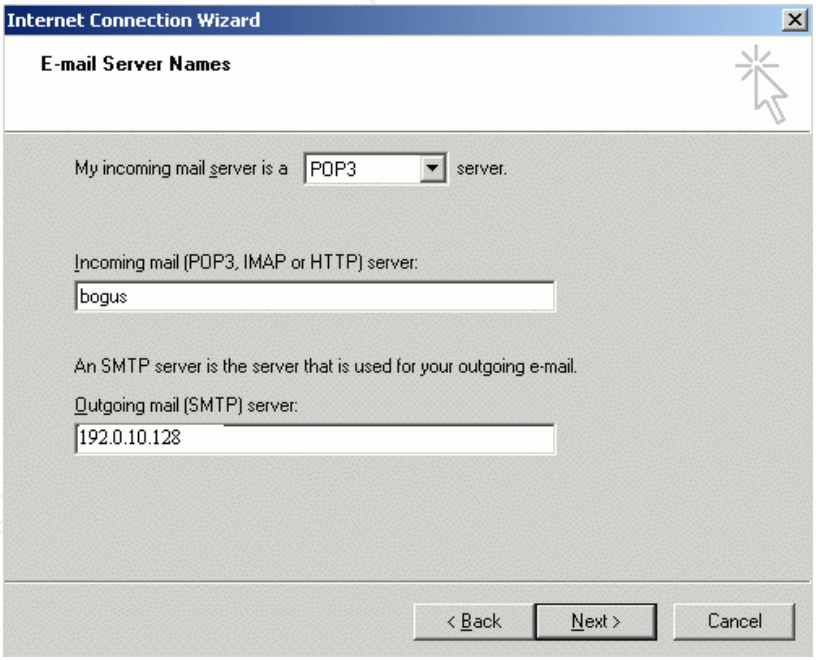

Figure 17: Configuring Outlook Express

After temporarily disabling his anti-virus software, Brady Uses the Outlook Express profile he just created to forward a copy of the W32/Swen HTML e-mail to every Example.com e-mail address he harvested.

From the Example.com perspective, the attack will come in through an allowed channel – e-mail – so getting around the firewall is not an issue. Once the message has been received by an e-mail server, the anti-virus software will scan it for malicious code. Because anti-virus vendors have only come out with definitions in the last day, Brady is banking on some or all of the mail servers not having current definitions, allowing the message to get delivered to the targeted recipients. Brady is not sure whether or not Example.com has any Intrusion Detection Systems (IDS) or Intrusion Prevention Systems (IPS) deployed, but in his mind it is not really an issue. Given that W32/Swen is so new, and that it is just now starting to emerge as a real threat, it is unlikely that IDS/IPS signatures are deployed or even available.

Once the message is delivered to the end recipient, their anti-virus software will scan for malicious content. When his brother had worked for Example.com, Brady had played around with his laptop on a few occasions. Because of this, he knows that the Example.com standard build uses Windows 2000 Professional, Microsoft Office 2000, Outlook 2000 e-mail client, and the Symantec anti-virus client solution, configured to download definition updates every Wednesday evening. This was great news for Brady since the Symantec definition files that detected W32/Swen had not been released until this last Thursday. Unless Example.com users manually initiated a definition file update, their machines would not be able to detect W32/Swen. From here, Brady will rely on two methods of infection. First, users who do not have the MS01-020 patch applied will become infected simply by opening the message or viewing it in their Outlook preview pane. Since the patch is almost two years old, Brady is not very confident in this infection method, although it is possible. The second, and more likely infection method, is a user installing the infected executable that is masquerading as a patch. A few times over the last couple of years Eric had mentioned to Brady that Example.com had begun a user awareness program focused on information security, and that patching known system vulnerabilities had been a big topic. Brady is hoping that diligent users who have received this training will be duped into applying the bogus security patch.

KEEPING ACCESS

With this particular attack, there is no need for Brady to keep access. Brady simply wants to unleash the worm to create frustration and mayhem on the Example.com network. However, W32/Swen needs to keep access in order to successfully propagate. As discussed previously, W32/Swen places a copy of itself, using a randomly named filename, in the %Windrir% directory. Any time a

EXE, .SCR, .COM, .BAT, or .PIF files is executed, W32/Swen will launch itself. W32/Swen also attempts to connect to all network shares and copy itself into the Startup directory of any Windows based device. In IRC, W32/Swen will modify the script.ini file, causing the worm to be run in memory and upload copies of it to other IRC users. In P2P network, W32/Swen uses enticing files names and creates its own folder share to continue to attempt infection to other P2P network users. W32/Swen will also attempt to post infected messages to newsgroups. These newsgroups can either come from the locally infected machine, or by using a list of predefined newsgroups that W32/Swen comes coded for. Furthermore, W32/Swen will seek to stop any services that may adversely affect it, such as desktop firewalls and anti-virus software. In addition, W32/Swen deeply entrenches itself in the system registry (please reference the *Descriptions and Signatures of Attack* section), making it difficult to completely eradicate.

COVERING TRACKS

Brady removes the Outlook Express configuration profile he just created, deletes the infected e-mail, and empties his recycle bin. He momentarily considers reformatting his laptop hard drive and reinstalling the operating system, but ultimately decides the effort would not be worth it. This example is essentially a "dumb-criminal" case. Brady is relying on Example.com assuming W32/Swen came into their network through normal virus transmission means, not from a specific source with malicious intent.

Had Brady been more concerned about this, there are various means he could have used to cover his tracks and distance himself from the activity. One method would have been to not only forge the source domain in the email address, but to also spoof the source IP address to make it match a legitimate Microsoft address range. This is not a simple process however, and required more time and effort than Brady was willing to invest.

Another possible method of covering tracks would have been to attempt to modify or delete the log files on the mail server so they did not point directly back to Brady's dynamically assigned DSL public IP address. To accomplish this, Brady could have used a tool such as Nessus to scan the Example.com mail server for any vulnerability that could be exploited to gain access to the log files.

In addition, W32/Swen makes its own attempt at covering tracks. If a user is tricked into filling out the bogus MAPI32 error that is displayed, W32/Swen will take this information and log into the POP3 server to check the email. If the W32/Swen finds an email that it sent, it will be deleted. W32/Swen will only delete the messages which the currently infected computer has sent.

5. THE INCIDENT HANDLING PROCESS

PREPARATION

Existing Countermeasures
Example.com has existing countermeasures in place that will aid in the handling of this incident.

- User Awareness Programs

Example.com has been diligently rolling out user awareness programs over the past year. The program consists of live training sessions on site, posters, and web-based training employees are required to take annually. The goal is to ensure every employee feels personally responsible for the security of Example.com's IT assets.

- Client Anti-Virus Solutions

The client anti-virus team is responsible for ensuring that every system connected to the network is running a current version of Norton AntiVirus that is enabled and has the most recent virus definition files applied. They work with the client delivery team to ensure that the software and updates are pushed out.

- Server Anti-Virus Solutions

The server anti-virus team is responsible for ensuring that all servers (file, web, mail etc.) have anti-virus solution deployed.

- Patch Compliance Standards

The threat assessment team is responsible for investigating new and emerging threats that could potentially affect the installation base at Example.com. Threats that could potentially have an adverse impact are ranked, and corresponding compliance timelines are derived from these rankings. The compliance timelines require users and delivery organizations to take the necessary action (patching, upgrading, etc) by the specified date.

- Automated Patching Process

The client delivery team is responsible for developing automated patch installations when possible. Depending on the threat assessment team's compliance time frame, the automated install can occur at logon/logoff, during non-business hours, or immediately.

Established Incident Handling Process

There is an established incident handling process. Example.com has a full-time, dedicated incident management team, made up of members in key areas such as anti-virus, threat assessment, vulnerability detection, counter measures, investigations, and incident response. The incident management team has members worldwide and uses a follow-the-sun operational model that ensures someone is available 24x7. Regions and coverage hours are broken down as follows:

Region	Coverage Times (in PST)
Americas region	8:00 – 16:00
Europe, Middle East, & Asia	16:00 – 24:00
Asia Pacific	24:00 – 08:00

This ensures that someone is always available to handle incidents as they occur. Staffing is specifically structured so that incident response team members and investigation team members are located in each region. This ensures that a qualified individual is always on duty and available. The investigations team is used as the point of contact for any incidents that may have legal repercussions. When an incident handler receives a case they believe may have potential legal impact, they will hand it off to the investigations team for consideration. Issues that will warrant a hand-off to the investigations team are incidents that involve fraud or potential fraud, real or potential monetary impact to the company, employee harassment, any incident that may have a human resources impact (suspension, termination, etc.), and incidents that clearly violate the law (child pornography for example). The investigations team will then assess the incident and either pass it back to the incident response team with handling instructions, work the case themselves, or involve appropriate internal and external legal assistance as needed. Internal legal assistance would be in the form of the human resource department or the in-house legal staff to assist in the handling of employee discipline for computer related abuse. External legal assistance would be any third party legal entity that would need to be involved to assist in the resolution of an incident, such as a high-dollar amount theft.

Groups, departments, and individuals within Example.com have been made aware of the incident management team's existence through awareness training consisting of on-site training classes, web-based tutorials. The Corporate Incident Response Team (CIRT) serves as the entry point for the user community to submit a request to the incident management team. Reporting parties can submit incidents to CIRT by calling the incident hotline, sending an e-mail sent to cirt@example.com, or by filling out a web form on the internal CIRT web site.

When CIRT receives a report, the incident is logged in the CIRT database via a web form. A variety of information is required to be tracked, including:

- Reporting party name and contact information
- Affected business unit
- Whether the source of the incident is internal or external
- Time the incident occurred
- Time the incident was reported to CIRT
- Time that CIRT created the case
- Relevant details of incident, such as:
  - IP addresses, system names that are involved
  - Relevant output of any queries (tracert, nmap, nbtstat, etc.)
- Record of CIRT's response
  - What CIRT did
  - Who was contacted and when
  - Who the incident was handed-off or escalated to (if needed)

If any required information has not been provided, the CIRT member on duty will contact the reporting party to obtain it.

When the case is entered, a unique identifier (case #) is assigned to the case. All cases follow the name format of CIRT#[4-digit Fiscal Year][5-digit Incrementing Number]. So for example, the 150th case created in fiscal year 2004 would have the identifier CIRT#200400105. This case number is then sent to the reporting party with an acknowledgement and any necessary instructions. Within the CIRT mailbox (Outlook client) a case folder (named after the case number) is created, and all case correspondence is stored in it. The CIRT mailbox and CIRT database are shared by all team members worldwide, so that follow up is possible on any given case 24-hours a day. The database and e-mail box are backed up daily, with offsite archives stored and kept for a minimum of five years, which is in keeping with Example.com's retention policy.

After the CIRT member has received an incident, created a case, and replied to the reporting party, they are ready to work the incident. The first step is to assess the incident to determine if it needs to addressed immediately or if it can take a back seat to any other pressing issues the CIRT member is currently handling. Incidents such as spam or a single virus report can be put off, while reports of an Example.com branded website defacement or a widespread denial of service (DoS) will take immediate priority. In any incident, the CIRT member's job is to gather the vital information as quickly and accurately as possible, communicate with the involved parties to ensure no unauthorized action takes place, and engage the appropriate teams that will be needed to resolve the incident. During the handling of an incident, the CIRT member is responsible for keeping the case record in the database updated and accurate. When the case is resolved, the CIRT member is responsible for closing the case in the database and archiving the case folder in the CIRT e-mail box. In addition, cases of a critical nature

(such as the website defacement or DoS mentioned previously) require a post mortem to be conducted. The CIRT member who owns the case is responsible for scheduling and conducting the post mortem.

Policy and Procedure Examples

There are many security policies in place at Example.com. The one that is most germane to the nature of this incident is the Anti-Virus Policy. Excerpts from the policy are provided below.

- General Anti-Virus Policy
    - All platforms covered by this standard must meet the following minimum requirements:
        - The capability to update virus definitions weekly for systems connected to Example.com network.
        - Automated weekly scanning of local storage containing Windows files.
        - The ability for mobile, remote, and home users to connect to a trusted virus definition server when the Example.com network is unavailable.

- Microsoft Windows Specific Anti-Virus Policy for Standard Image
    - Example.com Microsoft Windows based images must satisfy generic security requirements above and must also provide:
        - Automated alert notification to a centralized internal corporate 'alert server'.
        - The ability to push emergency updates of virus definitions when required to all systems connected to the Example.com network.
        - Real time protection provided for both local storage and e-mail.
        - Ability to lock virus software parameters, features, and settings in a manner that is not user changeable.

- Microsoft Windows Specific Anti-Virus Policy for Non-Standard Images
    - Windows operating system environments not running an Example.com 'standard' image must meet the generic requirements and:
        - The ability to receive and install emergency updates of virus definitions when required .
        - Real time protection provided for both local storage and e-mail.

- Microsoft Exchange Server Anti-Virus Policy
    - Microsoft Exchange anti-virus software must satisfy the generic security requirements and must be configured to the following minimum standards:
        - Automated alert notification to a centralized internal corporate 'alert server'.
        - The ability to push emergency updates of virus definitions when required.
        - Real time protection of local system files.
        - Microsoft Exchange servers must automatically scan all e-mail messages.
        - The anti-virus software must be configured to automatically check for virus detection file updates frequently.
        - All dangerous e-mail attachments must be blocked.

IDENTIFICATION

What follows is a scripted example describing how W32/Swen impacted one user, the users call to the help desk, and the help desk's subsequent escalation to the incident management team. In a large environment that had been impacted by W32/Swen, there would have several manifestations of the worm's propagation. Clearly, anyone who ran the infected executable on a machine that did not have appropriate virus definition files in place would become infected. These infected machines would then attempt to infect others by sending out

infected e-mails to all addresses they harvested on the infected machine, and by attempting to copy the worm into the startup folder of any Windows machines that were mapped via network shares from the infected machine. End users would quickly feel this impact in the form of network congestion as W32/Swen would be eating up resources through e-mail and network share traffic. All employees would begin to see multiple W32/Swen emails from infected employees. Infected employees would also see the MAPI32 Exception pop up window (please reference *Figure*) that frequently appears on infected machines. From a administration perspective, network managers would see the spike in bandwidth consumption caused by the W32/Swen's activities and e-mail server administrators would see the rapid rise in e-mail activity cause by W32/Swen's propagation technique.

❖ Monday, 22 September 2003
6:00am EDT

Sam Woods enjoyed his job at Example.com. He found the work challenging, and viewed himself as a loyal employee. As he started his Monday morning routine, he began sifting through the e-mails that had collected over the weekend. With one hand on the mouse, and one finger on the delete key, he was making quick work of it until he noticed a serious looking e-mail from security@microsoft.com. Reading the e-mail made him recall an IT security awareness briefing he had attended a few months back. The basic crux of the training had been that many of the IT security incidents that affected the company would have been non-issues if users had patched their systems. With this in mind, Sam launched the patch46.exe executable attachment, and continued on with his morning routine with a feeling of satisfaction.

❖ Monday, 22 September 2003
10:30am EDT

Joe Ridner hung up the phone with a sigh. 2½ hours into his help desk shift, it was becoming apparent that all was not right in the world. He had received numerous calls from users asking about a mysterious Microsoft security bulletin they had received. After conferring with his supervisor, they both agreed they needed to alert (CIRT).

❖ Monday, 22 September 2003
10:35am EDT

Bill Myers answered the CIRT hotline. After hearing Joe's report, Bill indicated that CIRT had also started to receive several user submissions about the same issue. Bill had already engaged the corporate anti-virus team, and they were

treating the attachment as malicious code. He gave Joe the incident identifier number to use for future Help Desk submissions or queries about this incident.

❖ Monday, 22 September 2003
10:36am EDT

Things were never boring on the anti-virus team. At least that is what Mike Callahan told everyone. As soon as CIRT forward him a copy of the bogus Microsoft security bulletin, Mike recognized it as W32/Swen, which had been identified by various anti-virus vendors the previous Thursday. Knowing that anti-virus definition files that would have caught this worm had been available since last Thursday afternoon, Mike gave a shake of his head and began making phone calls.

CONTAINMENT

By 12:00pm EDT, communication was e-mailed to all of the users in the company, alerting them to the W32/Swen virus activity, and instructing them to not open the attachment. Users were told that updated virus definition files would be pushed out to their computers within the next few hours. Some users would receive the definition files much sooner than that, but do to the network latency caused by W32/Swen activity, it was estimated that it would take 2-3 hours to push the definition files out to all connected users worldwide. It further instructed users to unplug their computers from the network and contact their local help desk if they believed they might be infected. This same communication was also posted on the main page of the Example.com Intranet site. It was also decided to send a company wide voicemail containing this information, since it was probable that some users would not be able to receive this communication via e-mail or the Intranet due to network congestion caused by W32/Swen.

The actual text of the message in the e-mail to employees and on the Intranet was as follows:

```
Early this morning in the Americas region, a virus
called W32/Swen infected several computers and began
to spread to other computers within Example.com
worldwide. Within a short amount of time, adverse
affects began to be felt by all users.

W32/Swen comes packaged and disguised as a security
update from Microsoft. It is critical that you DO NOT
open the attachment, as this will cause your computer
to become infected.
```

Updated definition files for Symantec Anti-Virus are currently being deployed. This will happen silently and automatically on you machine sometime within the next three hours. The correct virus definition files will be dated September 18th, 2003.

If you have already opened the attachment, or believe you are infected, please unplug your network cable immediately and call your local help desk. As you remember from the mandatory awareness training every employee has taken this year, the network cable is the cable plugged into the back of your computer or docking station that looks like a fat phone cord.

More information on W32/Swen can be found on our internal AV website:

http://Intranet.example.com/av

In addition, the W32/Swen removal instructions and tool can be found at:

http://securityresponse.symantec.com/avcenter/venc/data/w32.swen.a@mm.removal.tool.html

Thank you for your assistance with this issue. We anticipate a return to normal network activity by early this afternoon.

Best Regards,
Example.com IT Management

CIRT had been able to compile the current submissions, and the worm was pretty much limited to North American sites. There were a few reports out of Europe, but if they could act quickly, they could stop the spread before Asia-Pacific started their workday. The worm did not appear to be damaging anything directly, but it was creating chaos.

CIRT contacted Mike to let him know that a user in Seattle, Sam Woods, after seeing the notice on the Intranet, had called the Help Desk to report that he had opened the attachment early that morning. Knowing that it was probable that Sam's was one of the first machines infected, Mike contacted the IT investigations team and asked them to make a copy of Sam's hard drive so it could be analyzed later. The IT Investigations team called Sam and instructed him to unplug the power cord on the back of the machine and not do anything else to the computer until they arrived. With their jump kit in hand, the IT

Investigations team headed off to visit Sam. The jump kit they used included many items, including the following:

Software

- Copying software (dd, Ghost)
- Forensic software (EnCase)
- Sniffer software (ethereal)
- Windows resource kit
- Bootable OS floppy
- Disk drive duplicator (Logicube Omniclone Echo [17] – handheld)

Hardware

- Blank SCSI drive
- Blank IDE drive
- Small 4-port hub
- Cables (Cat5 patch, Cat5 crossover, coax, AUI, serial, rolled)
- Dual boot laptop (Windows 2000/RedHat)
- Screwdriver set
- USB pocket drive
- PCMCIA WiFi card
- Digital camera
- Flashlight

Miscellaneous Supplies

- Blank floppies, CD-R, tapes
- Pens and paper
- Cell phone with extra batteries
- Team and company contact lists
- Tape recorder
- Extra copies of all forms
- Evidence bags and ties

When they arrive at Sam's location they used the Logicube disk duplicator to make a bit for bit copy of the hard drive onto a new hard drive. The Logicube device is very straightforward to use – simply plug in the source drive on one end and the destination drive on the other, then use the control panel to start a bit-for-bit copy from the source drive to the destination drive. The newly copied hard drive was then placed in an electrostatic bag which was then sealed and tied in an evidence bag with a chain of custody form attached to it, and signed into the IT investigations storage vault. The storage vault is a DM3420-3 FireKing [18] data-media safe that is used to store all investigation related media. It has 6.0 cubic feet of capacity, a UL class 125 3-hour fire rating, and a UL RSC burglary

rating. The hard dive will remain in protective storage for a minimum of 5 years, which is in keeping with the Example.com retention policy.

The IT Investigations team could have simply taken Sam's hard drive and submitted it into evidence. However, Sam had vital work on the hard drive that was not available anywhere else on the network. Given the precise nature of the bit-for-bit copy, the documented chain of custody, the robust storage safe, and the company retention policy, they felt comfortable with having a backup for potential future legal efforts.

Because he wanted a first hand look at the infection and clean-up process, Mike personally worked with Sam to assess and contain the incident on Sam's computer. First, Mike had Sam unplug his computer from the network to prevent further W32/Swen related traffic. The following screen shots were captured from Sam's desktop, confirming the presence of W32/Swen.

Evidence of W32/Swen Infection

- The infected attachment was named "patch46.exe". Examining the properties of this file confirmed its size (106,496 bytes) was consistent with the reported size of the worm (*Figure 18*)

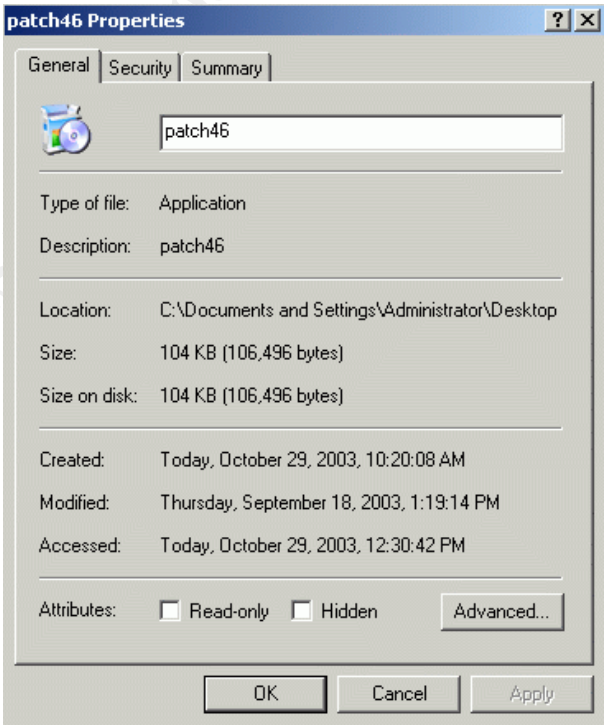

Figure 18: Byte size of W32/Swen payload

- Browsing of the WINNT directory show the presence of several files associated with W32/Swen:

    - Swen1.dat contains a list of 350 IP address and system names of remote mail and news servers. This file was in the C:WINNT directory, as shown in *Figure 19*.

    - The actual worm is dropped into the WINNT directory, named with a random string of letters. On Sam's machine, the worm was called syim.exe, as shown in *Figure 19*.

    - A batch file, named after the infected system's name, is also created. *Figure 19* shows the presence of "W2KA.bat" present in the WINNT directory. W2KA is the name of Sam's computer. The contents of this batch file are:

      ```
      @ECHO OFF
      IF NOT "%1"=="" syim.exe %1
      ```

      This code is used to launch the worm after it is dropped into the WINNT directory.

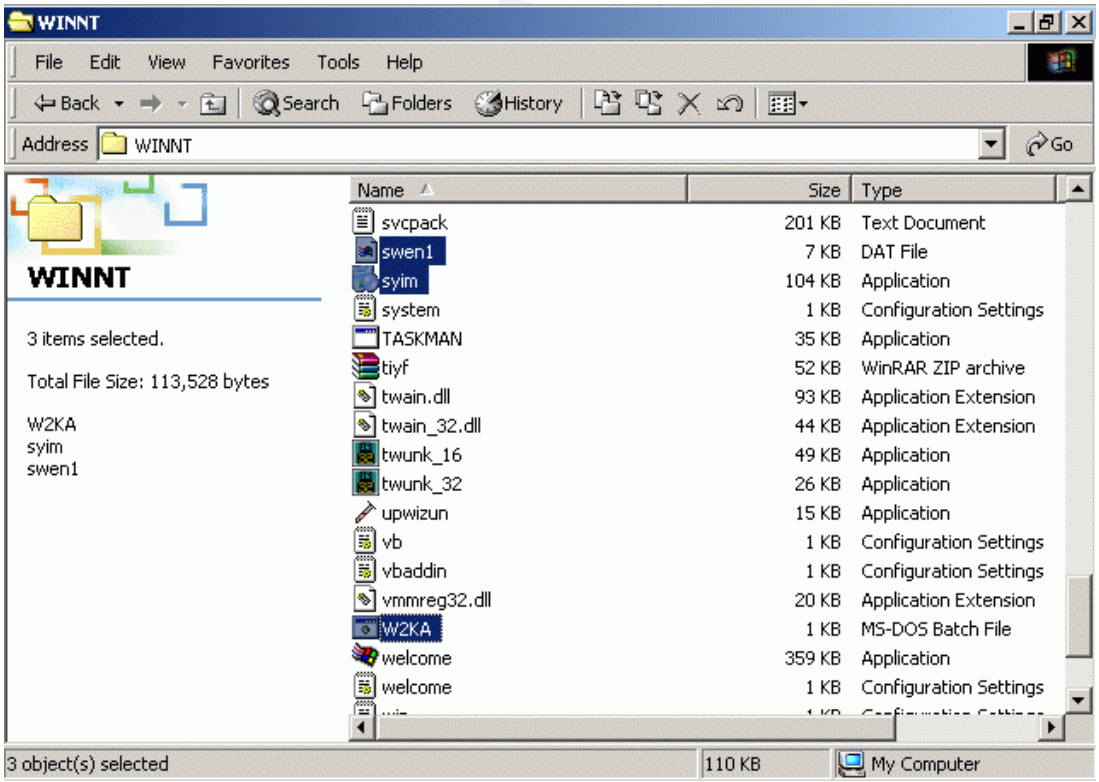

Figure 19: W32/Swen files in Winnt directory

- The frequent pop-up of the MAPI32 Exception error was also a indication of W32/Swen infection.

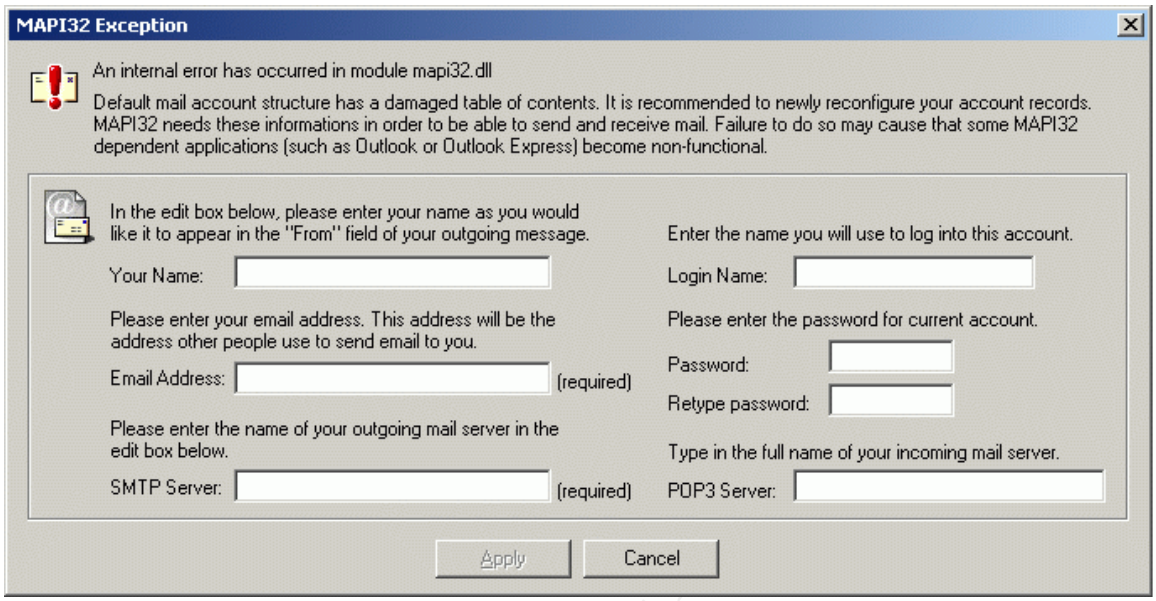

Figure 20: Bogus MAPI32 error generated by W32/Swen

- As a final test, Mike attempted to run the registry-editing tool Regedit. The resulting dialog box shown in *Figure 16* is also consistent with W32/Swen behavior.

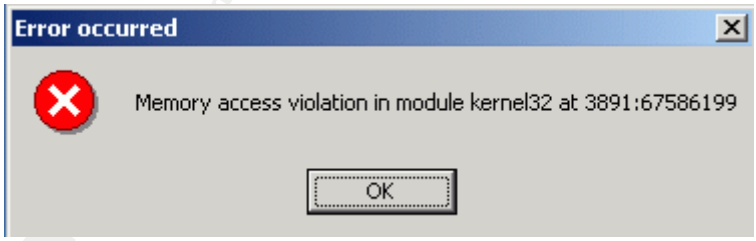

Figure 21: Error message when trying to launch process stopped by W32/swen

To analyze W32/Swen behavior and to ensure it was consistent with the behavior reported by the anti-virus community, Mike installed Windows 2000 Professional using the application VM Ware (www.vmware.com). VM Ware allows multiple operating systems to be installed on a single PC, and allows for the operating systems to be staged in virtual isolation for testing and development purposes. After installing Windows 2000 Professional in the isolated virtual environment, Mike copied the infected patch46.exe file to the virtual installation and ran it. By comparing the actual results with the results reported by the anti-virus vendors,

Mike was able to verify that there were not any apparent additional effects that needed to be corrected.

ERADICATION

2:00pm EDT

By now, Mike had contacted each of the relevant e-mail delivery organizations. They had all indicated they had applied the latest anti-virus definitions to the Exchange servers. The client anti-virus team said they had been able to push updated Symantec definitions out to all connected computers and that full system scans would be scheduled to run during the evening hours of each region. He had also sent a copy of the virus off to Symantec and they had verified that it was the same variant that had been reported last Thursday.

The threat assessment team verified that every user in the company had been migrated to IE 5.5 SP2 months earlier, and were not susceptible to the Internet Explorer MIME vulnerability discussed in MS01-020. The only way a user could become infected was by launching the executable or through the W32/Swen being copied via a network share.

Now that the problem had been contained, the cleanup efforts could begin. W32/Swen was difficult to clean manually, but fortunately Symantec offered an automated cleanup tool, FixSwen.exe [19].

Mike downloaded a copy of this tool onto a floppy disk and then ran it on Sam's computer. *Figure 22* shows the tool when launched.

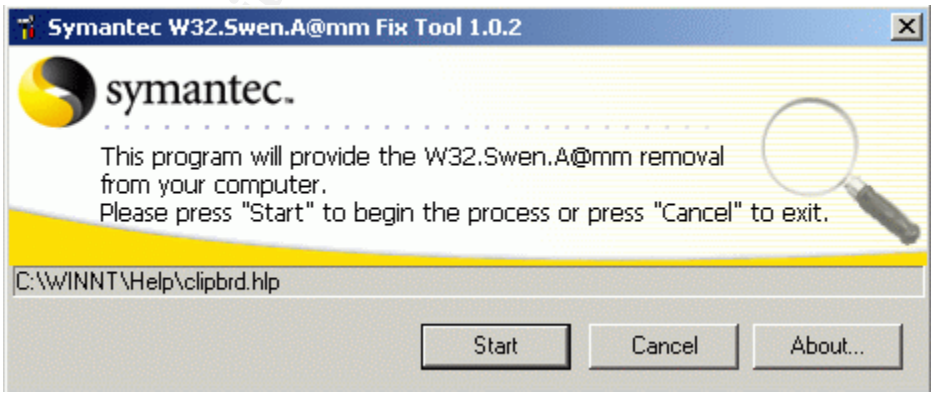

Figure 22: Symantec W32/Swen removal tool

Upon completion, the FixSwen.exe utility displayed the status message shown in *Figure 23*.

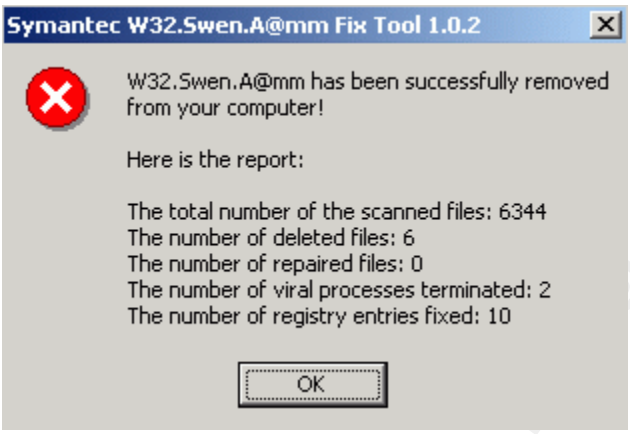

Figure 23: Symantec W32/Swen removal tool report

Even after the confirmation message from the cleanup utility, Mike still manually verified all of the registry keys and files had been removed or restored to their former state. He did this by examining the results from the FixSwen.log (generated by the FixSwen.exe utility). The log file contained the following:

```
<Start of FixSwen.log file>

The process "syim.exe" is viral. It has been
terminated.

Deleted the value "makwmxhcx" from the registry key
"HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Current
Version\Run".

The tool has deleted the file "C:\WINNT\uxpnyj.mqt".

Deleted the subkey "YHKANU" of the registry key
"HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\Current
Version\Explorer".

The value "DisableRegistryTools" of the registry key
"HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentV
ersion\Policies\System" is set to 0.

The default value of the registry key
"SOFTWARE\Classes\batfile\shell\open\command" is set
to ""%1" %*".
```

```
The default value of the registry
key"SOFTWARE\Classes\comfile\shell\open\command" is
set to ""%1" %*".

The default value of the registry key
"SOFTWARE\Classes\exefile\shell\open\command is set to
""%1" %*".

The default value of the registry key
"SOFTWARE\Classes\piffile\shell\open\command" is set
to ""%1" %*".

The default value of the registry key
"SOFTWARE\Classes\regfile\shell\open\command" is set
to "regedit.exe "%1"".

The default value of the registry key
"SOFTWARE\Classes\scrfile\shell\config\command" is set
to ""%1" %*".

The default value of the registry key
"SOFTWARE\Classes\scrfile\shell\open\command" is set
to ""%1" /S".

The tool has deleted the viral file "C:\Documents and
Settings\Administrator\Desktop\patch.EXE".

The tool has deleted the viral file
"C:\WINNT\swen1.dat".

The tool has deleted the viral file
"C:\WINNT\syim.exe".

The file "C:\WINNT\germs0.dbv" is infected. The file
has been deleted.

The file "C:\WINNT\W2KA.bat" is infected. The file has
been deleted.

W32.Swen.A@mm has been successfully removed
from your computer!

Here is the report:

The total number of the scanned files: 6344

The number of deleted files: 6
```

```
The number of repaired files: 0

The number of viral processes terminated: 2

The number of registry entries fixed: 10

<End of FixSwen.log file>
```

After taking these steps, Mike felt reasonably confident that W32/Swen had been cleaned from the machine.

Having someone visit every user desktop in the company to run the SwenFix tool was not a feasible option. The link to the cleanup tool, as well as basic use instructions, were e-mailed to the Example.com users, posted on the Example.com internal anti-virus site, and communicated to the help desk. End users were asked to run the SwenFix tool to ensure that W32/Swen had not infected their machines. The text of the message that appeared in the e-mail and on the Intranet was as follows:

> Earlier today, employees at Example.com were informed of the rapid spreading of a virus called W32/Swen. We are pleased to report that due to the timely response of our incident management team, the help desk, and Example.com employees, the negative impact is subsiding and network operations are returning to normal. At this time, update definition files have been pushed to all connected desktops. Desktops that are not currently connected will receive the update the moment they log on.
>
> As an added precaution, we would ask every employee to run an automated tool from Symantec, called SwenFix, which will scan your computer and remove any traces of W32/Swen it finds. The tool can be downloaded at:
>
> http://securityresponse.symantec.com/avcenter/venc/data/w32.swen.a@mm.removal.tool.html
>
> Simply save the SwenFix.exe file to your desktop, and double click to launch it. It will run quietly in the background, during which time you can continue your normal work. When it is finished, it will display a status screen letting you know the results.

```
If you have any questions about this process, please
contact your local help desk.

Best Regards,
Example.com IT Management Team
```

W32/Swen infected the Example.com network when Sam Woods had been tricked into opening an infected executable that was packaged as Microsoft security update. It had also slipped through Exchange 2000 e-mail servers that did not have current virus definition files applied. The root cause of this issue was a lack of user and team compliance with defined security policies.

RECOVERY

Once the FixSwen.exe cleanup utility was run on Sam's computer, Mike made sure that the Norton AntiVirus client (version 8.1.0.825, scan engine 4.2.0.7) service was enabled and configured to startup automatically. *Figure 24* shows the service is started and correctly configured.

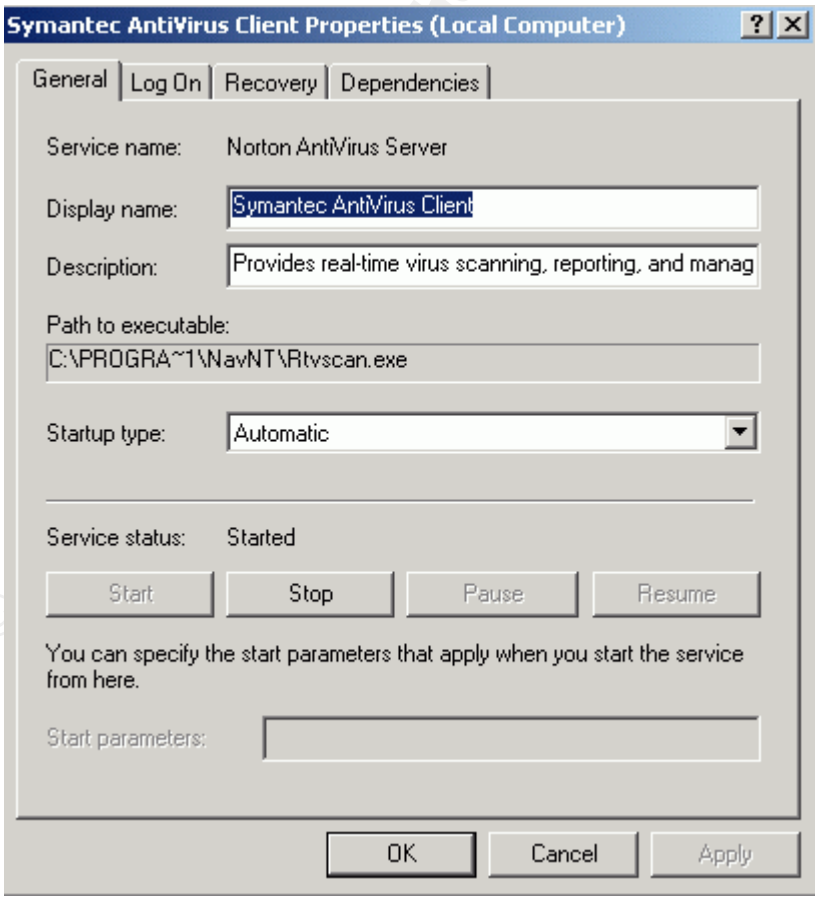

Figure 24: Symantec AV is running

Once this was confirmed, Mike verified that the virus definition files were dated 18 September 2003. He did this by double clicking on the Norton AntiVirus icon in the system tray and verifying the virus definition file version date in the bottom right hand corner. Then, Mike initiated a full system scan of the machine by clicking on the "Scan Computer" link in the left hand menu, selecting the drive to be scanned (in this case C:\), and then clicking on the "Scan" button in the lower right hand corner, as shown in *Figure 25*.

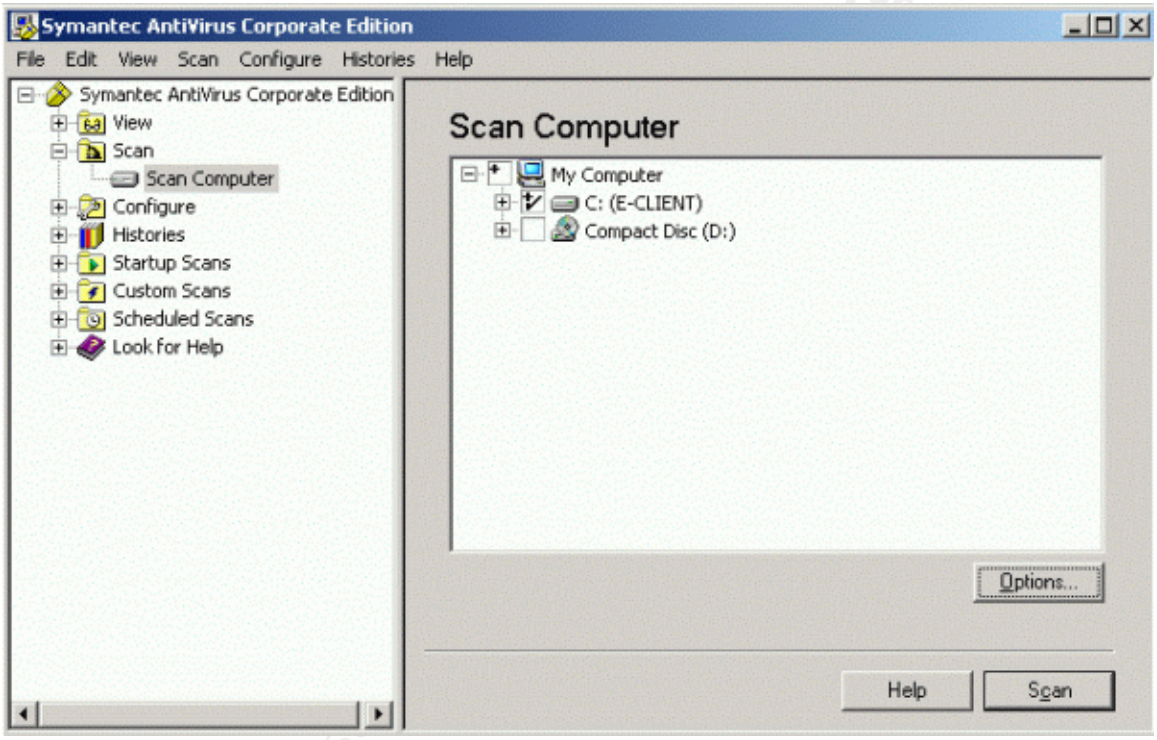

Figure 25: Using Symantec AntiVirus to scan a PC

The default scanning options were appropriate. When this scan came back clean, as shown in *Figure 26*, Mike felt confident that W32/Swen had been contained and eradicated from Sam's computer, and the recovery was complete on the system.

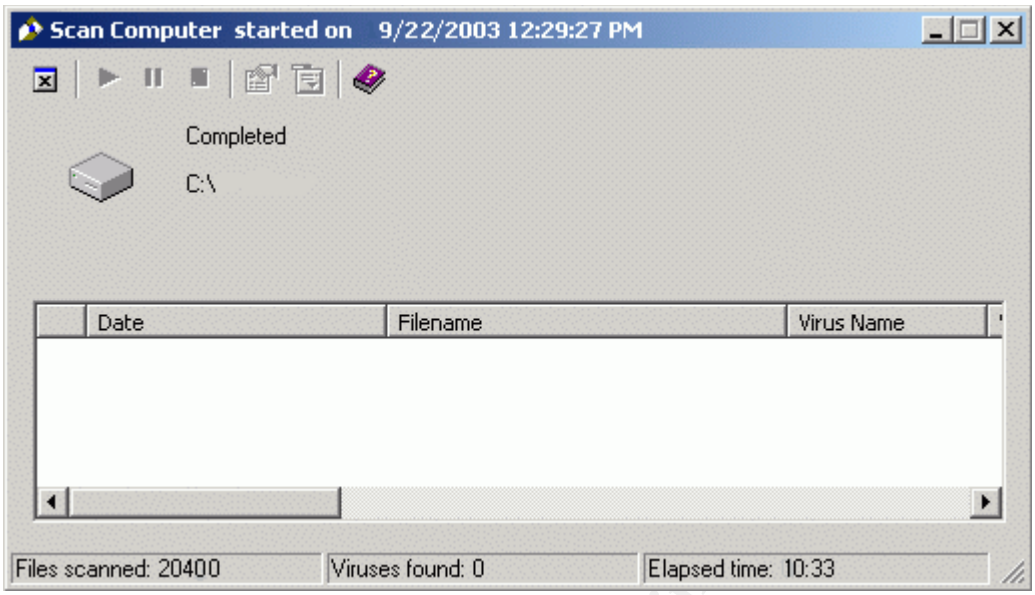

Figure 26: Report of a clean anti-virus scan

Apart from individual scans run on client computers by Symantec AntiVirus, Snort signatures were also deployed on internal sensors to detect any remaining W32/Swen traffic. Example.com devices that triggered Snort W32/Swen alerts were parsed out of the logs, and the help desk was tasked with following up with system owners to ensure proper cleaning steps were taken. The Snort signatures deployed to detect W32/Swen traffic were:

```
alert tcp $EXTERNAL_NET any -> any any (msg:"W32.Swen@mm -
SMB";content:"|59 59 85 C0 74 09 6A 01 58 83 4D FC FF EB 15 FF 85
E0 FE FF FF EB C7 6A 01 58 C3 8B 65 E8 83 4D|"; classtype:misc-
activity;rev:1;)

alert tcp $EXTERNAL_NET any -> any any (msg:"W32.Swen@mm -
MIME";content:"QABohKNAAGShAAAAAFBkiSUAAAAAgewUAQAAU1ZXiWXoM/+Jff
yJvdz +//+LdQhW6NORAABZhcB0"; classtype:misc-activity;rev:1;)
```

LESSONS LEARNED

By the next morning, reports of W32/Swen infections to the Help Desk and CIRT had slowed to a trickle. As Mike sat back and reviewed the events of the past 24 hours, several issues came to light.

The circumstances that caused Sam Woods to become infected were troubling. While the HTML e-mail was not a perfect Microsoft security bulletin, it was compelling. It was easy for Mike to see how it could prey on the good will of users who just wanted to do the right thing to protect IT assets. Trying to look at it from the perspective of a common end user, Mike realized what a confusing

landscape they navigated through. On one hand, the users were inundated with alerts and training that emphasized the importance of applying security patches in a timely manner. On the other hand, they must also be cautious and not open any e-mail or attachments that come from a suspicious or unknown source, no matter how legitimate they may appear.

Another troubling fact was that the definition files that would have detected W32/Swen were available at least three days prior to Example.com being infected with the worm. They had become accustomed to Symantec's Wednesday release of definition files, and all the clients connected to the corporate network were configured to update their definition files every Wednesday evening. They needed to establish a process to ensure that definition files release on other days of the week were pushed out to clients as soon as possible. There was no excuse for why the Exchange servers had not had current virus definition files applied.

On the following Monday (29 September 2003), the Example.com incident management team held a post mortem to discuss the events leading up to the W32/Swen outbreak and resulting lessons learned. Among the topics discussed were whether or not to pursue any legal action. In terms of business impact and cost, W32/Swen had resulted in approximately half a day's loss of productivity, mainly in the Americas region, and minimal costs associated with the clean up efforts. When compared to the costs incurred from Nimda and SQL Slammer, W32/Swen was barely a blip on the radar. Furthermore, W32/Swen had worldwide impact, and the infection source into Example.com could have come from an infected customer, supplier, etc. Even if someone had specifically targeted Example.com with W32/Swen, the damage was quickly contained and recovery was almost 100% complete. Based on this, the director of IT security at Example.com did not feel any legal action was financially justified. The other members of the incident management team supported this decision. However, if they had chosen to investigate this incident more fully, the first place would have been to start with a forensic examination of the duplicated hard drive in storage. It would have been a simple matter to retrieve the mail headers from the infected message. In Outlook, simply right click on the mail message then select Options to reveal the Message Options window that has the headers (as shown in *Figure 27*).

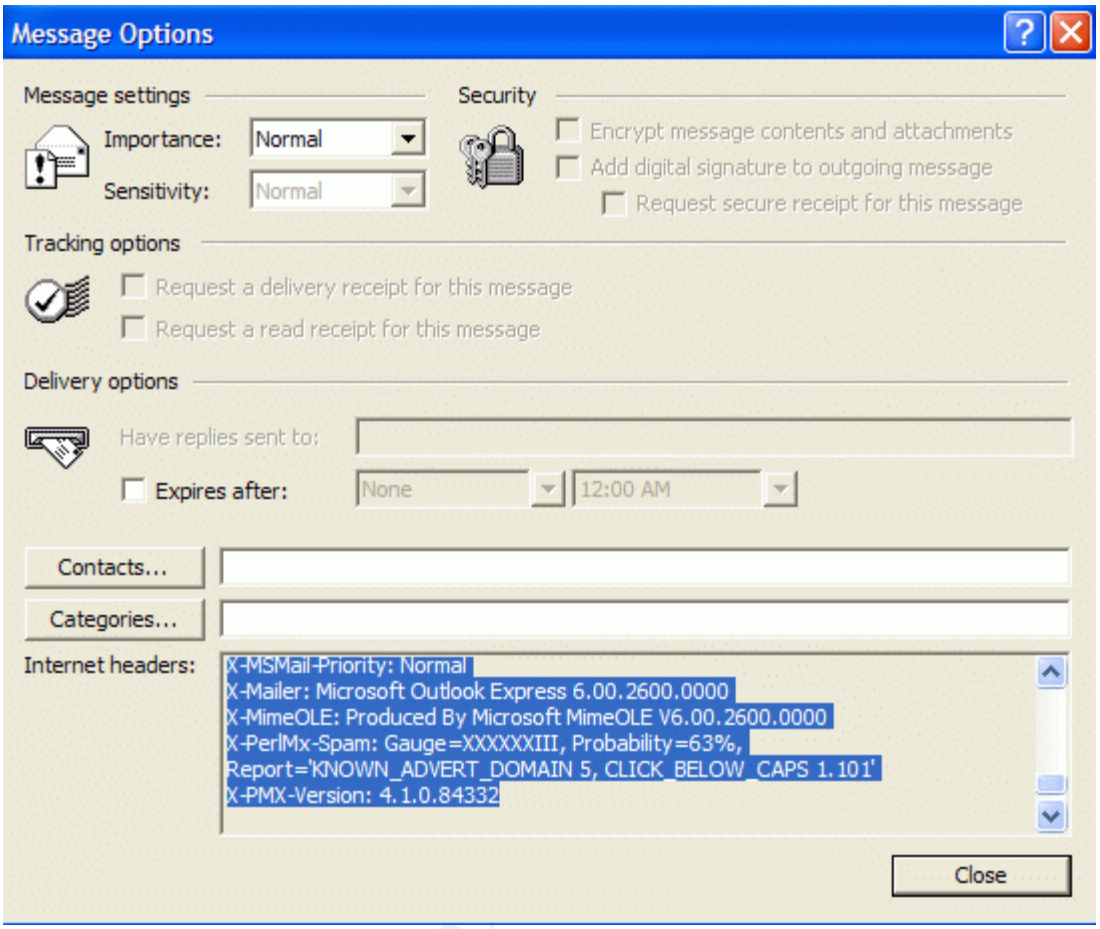

Figure 27: Viewing e-mail headers in Outlook

The header from the infected message would have looked something like:

```
Return-Path: <security@microsoft>

Received: some valid domain name [172.16.0.1] by mail.example.com
with SMTP id 1auBep2CR3Nl3qW0  Fri, 19 Sep 2003 20:32:27 -0500
(EST)

Received: from microsoft.com ([192.168.0.1])by some valid domain
name [172.16.0.1] with ESMTP id <705612-36520> Fri, 19 Sep 2003
20:31:55 -0500 (EST)
Message-ID: <3zci$3$-l3-6@j9k5amj.1qmb7>

From: "Microsoft Corporation <security@microsoft.com>

To: sam.woods@example.com

Subject: Latest Microsoft Security Patch

Date: Fri, 19 Sep 03 20:31:04 -0500 (EST)
```

```
X-Mailer: Microsoft Outlook Express 6.00.2600.0000
MIME-Version: 1.0
Content-Type: multipart/alternative;
boundary="60F_7576D85744B.E072E5"
X-Priority: 3
X-MSMail-Priority: Normal
```

Reading from the bottom up, we see *the date, subject, From:*, and *To:* addresses used by the message. The first received line indicates the sending party identified themselves has Microsoft.com. However, the resolved IP address of the sending mail server is 192.168.0.1, an address which is not owned or managed by Microsoft. From here the message is relayed to a mail server at *some valid domain name*. Then it is relayed to mail.example.com where it will be delivered to the mailbox of the intended recipient, assuming the destination e-mail address is correct. In the top received line, the mail server identifying itself as *some valid domain name* does correctly map to the resolved IP address of 172.16.0.1. The only glaring discrepancy in the e-mail header is apparent Microsoft.com forgery.

This information could have been compared with log files stored on the mail server to add further evidence. At this point, the Example.com legal team would be involved, and with the assistance of external law enforcement, and the ISP who owns the offending IP address, the attacker could be identified in short order.

In general, the incident management team handled the incident professionally and appropriately. Once the incident was escalated, action was efficient and effective. The necessary tools were available to the parties that needed them, and the jump kit allowed quick onsite response to what was identified as one of the initially infected machines.

One source of concern was the delay in the escalating the issue to the anti-virus team. The help desk alerted CIRT only after the W32/Swen reports had reached high levels. Thresholds need to be defined so that the help desk will escalate issues to CIRT before they get out of hand. Likewise, CIRT was somewhat tardy in engaging the anti-virus team. Had they involved them even an hour earlier, W32/Swen would have had less impact. It would have meant faster communication and quicker deployment of appropriate anti-virus definition files.

As a result of the meeting, the following executive summary and recommendations were reported to the incident management team director:

Executive Summary

On Monday morning, September 26th, Example.com employees began receiving copies of a worm known as W32/Swen. The worm comes packaged in an HTML e-mail, and is disguised as a security update from

Microsoft. The format of the e-mail is reasonably well done and preys on the good will of users who will install it thinking they are taking a proactive measure. A large number of users took this course of action Monday morning.

At the post-mortem held to analyze this incident, it was determined that if the current security policy had been followed, Example.com would not have suffered as much, if any, negative impact. As it was, normal business operations were interrupted for approximately 3000 users, and calls to the Help Desk and CIRT were above normal.

Virus definition files that would have detected and blocked W32/Swen were available on Thursday, September 18th. Due to poorly worded policy, the Norton AntiVirus software on the Exchange servers did not have the most recent definition files applied. Had they been, the worm would most likely not have found its way into the Example.com network. Client computers were on a weekly update schedule that was appropriate. The failure was in not identifying the virus definition files release last Thursday as critical and failing to push them out to the clients.

Recommendations

- The anti-virus team and threat team need to work together more closely to evaluate emerging malicious code. This will allow for an accurate determination to be made as to when an emergency virus definition files update needs to be pushed.

- The current policy governing installation of virus definition files on the Exchange servers needs to be changed. It currently states that definition files need to be updated "frequently". This direction is too arbitrary. The wording should be changes to state that definition files on the Exchange servers need to be updated hourly.

- The anti-virus team needs to work closely with the client delivery organizations to ensure that definition files are delivered in a timely manner. The goal is to have all network connected client computers updated within 24 hours of each definition file release.

- The anti-virus team needs to perform frequent system audits on Exchange
  e-mail servers to ensure that Norton AntiVirus is configured and functioning in a manner that is consistent with established security policy.

- User awareness training must continue to happen, and become a regular, repetitive course for all employees. The training needs to

strike a balance between encouraging employee compliance to protect IT assets and encouraging employees to be cautious when manually applying or installing any new program or patch.

- Thresholds need to be developed (in particular for the help desk and CIRT) so that emerging issues are escalated in a timely manner. This will allow for faster response and as a result minimize negative impact.

CITATIONS & REFERENCES

[1] Trend Micro. (2003). Glossary of Virus Terms. Retrieved September 20. 2003, from http://www.trendmicro.com/en/security/general/glossary/overview.htm#Worm

[2] Kleinsen, J. (2001). Simple Mail Transport Protocol. *RFC2821*. Retrieved November 7. 2003 from http://www.ietf.org/rfc/rfc2821.txt?number=2821.

[3] Oikarinen, J., & Reed, D. (1993). Internet Relay Chat Protocol. *RFC1459.* Retrieved September 21, 2003 from http://www.ietf.org/rfc/rfc1459.txt.

[4] CapnBry. Gnutella Protocol. Retrieved September 28, 2003 from http://www.capnbry.net/gnutella/protocol.php

[5] Wikipedia. (2003). FastTrack. Retrieved September 28, 2003 from http://en.wikipedia.org/wiki/FastTrack

[6] Kantor, B., & Lapsley, P. (1986) Network News Transfer Protocol. *RFC977*. Retreived October 1, 2003 from http://www.ietf.org/rfc/rfc0977.txt

[7] Symantec. (2003). W32.Swen.A@mm. Retrieved September 1, 2003 from http://securityresponse.symantec.com/avcenter/venc/data/w32.swen.a@mm.html

[8] Trend Micro. (2003). WORM_SWEN.A. Retrieved September 1, 2003 from http://www.trendmicro.com/vinfo/virusencyclo/default5.asp?VName=WORM_SWEN.A&VSect=T

[9] Sophos. (2003). W32/Gibe-F. Retrieved September 2, 2003 from http://sophos.com/virusinfo/analyses/w32gibef.html

[10] McAfee. (2003). W32/Swen@MM. Retrieved September 2, 2003 from http://us.mcafee.com/virusInfo/default.asp?id=description&virus_k=100662

[11] Krusesecurity. (2003). W32.Gibe-E.worm. Retrieved September 5, 2003 from http://www.krusesecurity.dk/advisories/gibe-e.txt

[12] Microsoft Corporation. (2003). Microsoft Security Bulletin MS01-020. Retrieved September 10, 2003 from http://www.microsoft.com/technet/treeview/default.asp?url=/technet/security/bulletin/MS01-020.asp

[13] CA. (2003). Win32.Swen.A. Retrieved September 15, 2003 from http://www3.ca.com/solutions/collateral.asp?CT=27081&CID=50601

[14] http://www.etree.org/md5com.html

[15] www.samspade.org

[16]http://www.foundstone.com/index.htm?subnav=resources/navigation.htm&subcontent=/resources/proddesc/superscan.htm

[17] http://www.logicube.com/

[18] http://www.firekingoffice.com/

[19]http://securityresponse.symantec.com/avcenter/venc/data/w32.swen.a@mm.removal.tool.html

Additional References

http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2001-0154

http://www.technewsworld.com/perl/story/31632.html

http://www.microsoft.com/security/antivirus/swen.asp

http://news.zdnet.co.uk/internet/security/0,39020375,39116520,00.htm

APPENDIX A: FIGURE REFERENCE

APPENDIX B: LIST OF REMOTE MAIL & NEWS SERVERS W32/SWEN REFERENCES

0.abnormal.com
12-252-202-62.client.attbi.com
12-254-107-9.client.attbi.com
140.109.13.17
141.4.4.45
141.44.21.70
142.155.129.4
161.53.2.66
165.84.1.12
192.83.173.60
193.54.76.35
194.133.33.10
194.204.186.44
195.208.172.9
195-241-98-179-mx.xdsl.tiscali.nl
200.68.14.42
202.102.2.93
202.108.36.140
202.123.213.31
202.153.122.217
202.184.155.10
202.85.146.123
202-177-16-121.kdd.net.hk
203.99.143.60
207.105.83.65
207.230.236.9
207.41.8.25
208.149.207.130
209.1.14.29
210.221.55.119
211.157.100.15
213.56.195.71
24.131.101.15
24.132.106.153
24.132.5.12
24.136.153.34
61.156.20.89
64.14.86.166
81.176.66.27
aboukir-101-1-4-pparis.adsl.nerim.net
acs2.byu.edu
adsl-82-64.36-151.net24.it
afpa01.gulliver.fr
ail-003.aifis.ne.jp

alpha.webusenet.com
argos.sae.gr
arroyo.abris13.org
asics.co.jp
aurelia.deine.net
baldrick.blic.net
baracka.rz.uni-augsburg.de
baran22.ttnet.net.tr
bastet.chchpoly.ac.nz
bbsnews.ndhu.edu.tw
beech.fernuni-hagen.de
bias.ipc.uni-tuebingen.de
bipnet.pl
blob.linuxfr.org
bolo.nais.com
bolzen.all.de
bolzen.logivision.net
bossix.informatik.uni-kiel.de
butthead.cybertrails.com
c3po.brook.edu
c66.190.46.194.fdl.wi.charter.com
cabale.usenet-fr.net
carbone.net.espci.fr
ccnews.thu.edu.tw
cdr.nord.net
colargol.tihlde.hist.no
concern.wolters-kluwer.nl
corp.newsgroups.com
corp-binaries.newsgroups.com
correo.uvigo.es
cowee.wcu.edu
cypress.alberni.net
davide.msoft.it
ddt.demos.su
decolores.net
dejasearch.wu-wien.ac.at
demonews.mindspring.com
diana.bcn.ttd.net
diesel.cu.mi.it
dns2.globalreach.net
dogwood.fernuni-hagen.de
dp-news.maxwell.syr.edu
dtv3.deltatelecom.ru
dvibm3.gkss.de
ep3000.scl.kyoto-u.ac.jp
etel.ru

fb1.euro.net
feed1.uncensored-news.com
feed2.uncensored-news.com
fguillien.net1.nerim.net
fido.almaty.idc.kz
fifi.woody.ch
flis.man.torun.pl
forums.novell.com
freebsd.csie.nctu.edu.tw
frmug.org
frmug-gw.frmug.org
ftp.tomica.ru
gail.ripco.com
gard.gigatrading.se
gatekeeper.laudaair.com
globo.edinfor.pt
glu08.dna.affrc.go.jp
graf.cs.uni-magdeburg.de
grapevine.lcs.mit.edu
grieg.uol.com.br
gsc.gsi.com
gwdu112.gwdg.de
h66-59-175-15.gtconnect.net
hermes1.rz.hs-bremen.de
host119-107.pool8172.interbusiness.it
htsrv.attack.ru
hub1.meganetnews.com
humbolt.nl.linux.org
charlebourg-1-81-57-17-164.fbx.proxad.net
chat.fibertel.com.ar
chinese.iie.ncku.edu.tw
chivato.uah.es
iis.tordata.se
inetgate.tp.ac.sg
info.rgv.net
info.tsu.ru
info4.uni-rostock.de
infosun2.rus.uni-stuttgart.de
inx3.inx.net
ip156.et.bocholt.fh-gelsenkirchen.de
isgnt5.netnow.net
l1.newaygo.mi.us
lace.colorado.edu
learnet.freenet.hut.fi
leda.omp.ad.jp
list.ege.edu.tr

lord.usenet-edu.net
lugnet.com
main.gmane.org
menuhin.netfront.net
miza.nu
mongol.sasknet.sk.ca
moon.ees.hokudai.ac.jp
mordor.jysnet.org
msnews.microsoft.com
mu-gateway.jasien.net
narzisse.hrz.tfh-wildau.de
natasha.ncag.edu
neptun.beotel.yu
netnews.de
news.abcs.com
news.ajou.ac.kr
news.aktrad.ru
news.aoc.gov
news.avcinc.com
news.avicenna.com
news.beta.kz
news.bsi.net.pl
news.caiwireless2.com
news.caravan.ru
news.caribsurf.com
news.cat.net.th
news.cdpa.nsysu.edu.tw
news.cell.ru
news.cofc.edu
news.coli.uni-sb.de
news.com2com.ru
news.comtel.ru
news.corvis.ru
news.cs.nthu.edu.tw
news.cs.tu-berlin.de
news.datast.net
news.deakin.edu.au
news.detnet.com
news.discom.net
news.dma.be
news.dna.affrc.go.jp
news.dsuper.net
news.emn.fr
news.enet.ru
news.freenet.de
news.fwi.com

news.fxalert.com
news.gamma.ru
news.gcip.net
news.gdbnet.ad.jp
news.globalpac.com
news.hanyang.ac.kr
news.htwm.de
news.ind.mh.se
news.inet.gr
news.informatik.uni-bremen.de
news.infotecs.ru
news.intel.com
news.invarnet.inwar.com.pl
news.isu.edu.tw
news.itcanada.com
news.jerseycape.net
news.kiev.sovam.com
news.konkuk.ac.kr
news.krs.ru
news.leivo.ru
news.lit.ru
news.louisa.net
news.lsumc.edu
news.lucky.net
news.man.torun.pl
news.math.cinvestav.mx
news.matnet.com
news.maxnet.ru
news.mc.ntu.edu.tw
news.mindvision.com.au
news.ncue.edu.tw
news.netcarrier.com
news.netdor.com
news.nchu.edu.tw
news.nsysu.edu.tw
news.odata.se
news.online.de
news.phoenixsoftware.com
news.portal.ru
news.primacom.net
news.ramlink.net
news.read.kpnqwest.net
news.readfreenews.net
news.reference.com
news.ripco.com
news.rt.ru

news.ru
news.ruhr-uni-bochum.de
news.savvis.net
news.sexzilla.com
news.solaris.ru
news.spiceroad.ne.jp
news.srv.cquest.utoronto.ca
news.sti.com.br
news.tehnicom.net
news.teleglobe.net
news.telepassport.de
news.terra-link.com
news.tln.lib.mi.us
news.tohgoku.or.jp
news.triax.com
news.ttnet.net.tr
news.tu-ilmenau.de
news.udel.edu
news.uncensored-news.com
news.uni-duisburg.de
news.uni-erlangen.de
news.uni-hohenheim.de
news.uni-mannheim.de
news.uni-rostock.de
news.uni-stuttgart.de
news.unitel.co.kr
news.univ-nantes.fr
news.utb.edu
news01.uni-trier.de
news1.sinica.edu.tw
news2.new-york.net
news4.euro.net
news4.odn.ne.jp
news4.uncensored-news.com
news-archive2.icm.edu.pl
newscache0.freenet.de
newscache1.freenet.de
newscache2.freenet.de
newscache3.freenet.de
newscache4.freenet.de
newscache5.freenet.de
newsfeed.ctrl-c.liu.se
newsfeed-west.nntpserver.com
news-read2.maxwell.syr.edu
news-rm.gamma.ru
newssvr20-ext.news.prodigy.com

nnrp-ham.news.is-europe.net
ns.alcatel.pt
ns.stirol.com.ua
ns2111.ovh.net
ns4.bih.net.ba
nserver.enc-1.com
nsnmpen2-lo.nuria.telefonica-data.net
oak.cise.ufl.edu
okapi.ict.pwr.wroc.pl
p59-2.choin.netsurf.de
pcp03428581pcs.waldlk01.mi.comcast.net
peabody.colorado.edu
peewee.greater.net
penelope-gw.oswego.edu
pg-adr-exch-01.adr.unbc.ca
plonk.apk.net
pluto.sm.dsi.unimi.it
pluto.srv.dsi.unimi.it
portraits.wsisiz.edu.pl
post.newsfeeds.com
pronews.centramedia.net
proxy.dvgd.ru
pubnews.gradwell.net
puce.geeks.org
pula.financenet.gov
pumba.class.udg.mx
ran.age.ne.jp
ratatosk.dvnc.net
rdr1.wms.teleglobe.net
rebell.ghks.de
regulus.its.deakin.edu.au
rock.afsac.wpafb.af.mil
rtcsrv5.realtech.de
rupert.mfn.org
s1.texinet.com
s216-232-127-148.bc.hsia.telus.net
sbs004.sitebuilder.at
selenium.club.cc.cmu.edu
server.internetoutlet.net
server.pspu.ac.ru
service.symantec.com
skarjeke.ind.mh.se
snews.apol.com.tw
snoopy.bndlg.de
sot-mod02.interalpha.net
sparky.midwest.net

sunsite.dk
sunu789.rz.ruhr-uni-bochum.de
supern2.lnk.telstra.net
tabloid.uwaterloo.ca
talia.mad.ttd.net
targetvision.com
tcr-04-14.pdx.du.teleport.com
test.easynews.com
tiger.aba.net.au
tindur.vks.is
tomcat.admin.navo.hpc.mil
tomcat.med.uoeh-u.ac.jp
traffic.uncensored-news.com
tthsc5.ttuhsc.edu
tyr.eiknes.se
ufik.idn.org.pl
ultra60.mat.uni.torun.pl
vanaema.matti.ee
vulkan.euv-frankfurt-o.de
weber.techno-link.com
welch-bm.welchandco.ca
wisipc.weizmann.ac.il
wixer.greyware.com
wixer052.greyware.com
www.an.cc.mn.us
www.focalnet.com
www.siast.sk.ca
www.usenet.pl
yellow.geeks.org
yucatan.franconews.org