48

VIRUS SYMPTOMS AND COUNTERMEASURES

CHAPTER AT A GLANCE

While most of the software products in the marketplace today are useful, constructive, and beneficial, there is also other software that serves a darker purpose—the *computer virus.* Such rogue software is designed to load and run without the user's knowledge, often hiding in normal programs. Viruses execute their functions without prompting users for permission, and they neither warn of potential dangers to the system nor produce error messages when problems are encountered. Essentially, a computer virus is a fragment of executable code that runs secretly and is capable of cloning itself in other programs.

Technically, there is nothing in this definition to indicate that a virus is necessarily *destructive*—that's a twist added by the virus programmers themselves. But legitimate software does not *need* to

run secretly, hide itself in other programs, or duplicate itself without a user's knowledge or permission. So the very nature of a computer virus makes it an ideal vehicle for spreading computer chaos—and in this "connected" world, computer viruses are more dangerous than ever. This chapter is intended to explain the nature and operations of computer viruses, show you how they spread and manifest themselves, and explain some procedures you can take to protect yourself and your customers from their effects.

Understanding Virulent Software

We use the term "virus" to describe virtually any type of destructive software. Although this is a good, general term, it is also a misnomer—a virus is actually only one of many destructive software types. There are at least nine types of recognized rogue software, and most are considered every bit as deadly as a virus. Each type of software has a different mode of operation. As a technician, you should understand how these classic software types operate.

SOFTWARE BUGS

Simply speaking, a *software bug* is an error in program coding or logic that results in faulty or unexpected operation. Bugs are rarely intentional, and the vast majority of *serious* system-crippling bugs are caught during the developer's alpha and beta testing processes. In order for serious bugs to get through into a finished product (the kind of bugs that can cause serious memory errors or damage hard drive files), the developer would have to do little (if any) testing on various PC platforms. Serious bugs are typically not intended as malicious, but they suggest a dangerous lack of concern on the part of the software developer. There are two clues that suggest the presence of software bugs: first, it is only a single program (usually the one you just installed or started using) that causes the problem, and the problem will not be detected by any antivirus tool (the application will be reported as clean). Software containing serious or persistent bugs is often referred to as *bug ware*.

TROJAN HORSES

The *Trojan horse* is largely considered to be the grandparent of today's virulent software. Basically, the Trojan horse is a destructive computer program concealed in the guise of a useful, run-of-the-mill program, such as a word processor or graphics program. Well-developed user shells or seemingly normal operations trick the user into believing that the program is harmless—until the virulent code is triggered and the program's *true* nature is revealed. The Trojan horse tactic is the most popular means of introducing viruses by distributing seemingly harmless software that actually contains virulent code. Fortunately, most virulent code can be detected by scanning new software before it is executed for the first time. To prevent the spread of Trojan horses, be suspicious of unwanted or unsolicited software arriving through the mail or as e-mail attachments. Also beware of software that sounds *too good* to be true (for instance, a TSR that will increase Windows performance by 100X, get SVGA graphics on an EGA video adapter, use AOL for free, and so on).

SOFTWARE CHAMELEONS

Just as a chameleon hides itself by mimicking its background, *software chameleons* mask virulent code with an image of a legitimate application. Of course, the mask is just a facade—like a demonstration program or a simulation. What makes a chameleon different from a Trojan horse is that *it almost never causes*

system damage. Instead, it generally makes a modification to a program. In one classic case, a chameleon was introduced to a large multi-user platform. When users typed in their name and password, they were recorded to a secret file. The chameleon's author later accessed the system, entered his own code, and downloaded the accumulated list of passwords. Thus, the author now had access to the users' data for his own illegal purposes. In another case, a chameleon was planted into a banking program that automatically diverted a few tenths of a cent (round-offs) from every transaction into a secret account. Ultimately, the chameleon's author had amassed hundreds of thousands of dollars in the secret account.

SOFTWARE BOMBS

The *software bomb* is just what the name implies—when the infected program is launched, the virulent "bomb" code executes almost immediately and does its damage. Software bombs typically contain no bells or whistles—they also make little effort to cloak themselves and almost no effort to replicate. The software bomb may, therefore, be developed quickly and easily. Their somewhat clumsy nature also makes them fairly easy to spot with antivirus tools.

LOGIC BOMBS

Where the software bomb is used for immediate and indiscriminate destruction, a *logic bomb* is set to go off when a particular logical condition is met. For example, the logic bomb may "detonate" (erase files, calculate subsequent payroll records incorrectly, reformat the disk, and so on) if payroll records indicate that the bomb's author is fired or laid off, or if the author's payroll statements do not appear for over four weeks. A logic bomb can be triggered by virtually any system condition. However, the "bomb" approach is fairly easy to spot with antivirus techniques.

TIME BOMBS

Instead of triggering a bomb immediately or through system status conditions, a *time bomb* uses time or repetition conditionals. For example, a time bomb can be set to "detonate" after some number of program runs, on a particular day (for instance, April 1st or Friday the 13th), or at a certain time (for instance, midnight). Time bombs are often used as a means of "making a statement" about a particular date and time. This kind of bomb architecture is relatively easy to spot with antivirus tools. Table 48-1 lists the activation dates of many known "traditional" computer viruses, while Table 48-2 lists a typical 12-month calendar of modern virus "payload dates." You can research current activation dates at **http://www.mcafee.com/centers/anti-virus/calendar/default.asp**. The advantage of a virus calendar is that you can quickly reference potential virus problems, then search for more detailed information from a virus library, such as http://vil.mcafee.com/.

TABLE 48-1 ACTIVATION DATES OF MANY "TRADITIONAL" COMPUTER VIRUSES

ACTIVATION DATE/DAY	VIRUS NAME
Sundays (any)	Mindless Sunday Sunday-2 Witcode
Sundays After 9th (Apr—Dec)	Doctor Qumak 2

TABLE 48-1 ACTIVATION DATES OF MANY "TRADITIONAL" COMPUTER VIRUSES *(CONTINUED)*

ACTIVATION DATE/DAY	VIRUS NAME
Mondays (any)	Carfield I-B (BadGuy) I-B (BadGuy 2) I-B (Exterminator) Immolation Kalah (Kalah-499) Witcode
Mondays (starting in 1993)	VirDem (VirDem-833)
Monday first of month	Beware
Mondays the 28ths	Crazy Eddie
Tuesdays (any)	Ah Emo-899 I-B (Demon) I-B (Demon-B) Murphy (Kamasya)
Tuesday The 1st	Jerusalem (JVT1)
Tuesday The 13th	Jerusalem (Anarkia)
Wednesdays (any)	PS-MPC (No Wednesday) VCL (Red Team) Victor
Thursdays (any)	TPE (Girafe)
Thursday The 12ths	CD
Fridays (any)	Bryansk Immolation Frere Jacques PS-MPC (Mimic-Den Zuk) PS-MPC (Mimic-Jerusalem) Murphy (Smack) NaziPhobia TalkingHeads VCL (Diarrhea) Wild Thing 2
Friday Not The 13ths	Jerusalem (Payday)
Friday The 11ths	VCL (Kinison)
Friday The 13ths	1720 Friday 13th Jerusalem RAM Virus Suriv 3.00 Westwood Witcode
Friday The 13ths (starting in 1992)	Hybryd

TABLE 48-1 ACTIVATION DATES OF MANY "TRADITIONAL" COMPUTER VIRUSES *(CONTINUED)*

ACTIVATION DATE/DAY	VIRUS NAME
Fridays After 15th of month	Jerusalem (Skism) Jerusalem (Skism-1)
Fridays last of month	Jerusalem (Sub-Zero B)
Saturdays (any)	Murphy (Finger) Jerusalem (Phenome) Murphy (Migram)
Saturday the 14ths	Saturday The 14th
1st day of any month	10 Past 3 Pinworm
2nd day of any month	Flip Tormentor (Nuke)
3rd day of any month	VCL (Miles)
5th day of any month	Frogs
7th day of any month	Bones
8th day of any month	Taiwan
10th day of any month	Day10 Leprosy (Leprosy-664A)
13th day of any month	NPox (NPox 2.1) Monxla Rocko
16th day of any month	10 Past 3
18th day of any month	Npox FORM-Virus (Form-18)
20th day of any month	Day10
22nd day of any month	10 Past 3 VCL (Beva 32)
24th day of any month	FORM-Virus Rocko (Mutating Rocko)
29th day of any month	10 Past 3 Geek Highlander
30th day of any month	Day10
31st day of any month	Tormentor (Lixo Nuke) VCL (Diogenes)
January 1st	Big Bang VCL (Beva 33)
January 1st—September 21st	Plastique (COBOL)
January 5th	Barrotes Joshi
January 15th	Casino
January 25th	Jerusalem (January 25th)

TABLE 48-1 ACTIVATION DATES OF MANY "TRADITIONAL" COMPUTER VIRUSES *(CONTINUED)*

ACTIVATION DATE/DAY	VIRUS NAME
February 1st—February 29th	Vienna (Beta Boys)
February 2nd	Dark Avenger (Amilia) Marauder
February 23rd	Swedish Boys (Why Windows)
February 24th	Swedish Boys (Why Windows)
February 25th	Swedish Boys (Why Windows)
February 28th	Zaphod
March 1st—March 31st	Fich Micropox
March 5th	X-2 (X-1 & X-1B)
March 6th	Mich II Michelangelo RIP-699
March 14th	Arale
March 15th	Maltese Amoeba
March 25th	March 25th
March 31st—April 30th	Mordor.1110
April 1st	Casper Christmas Tree Suriv 1.01 Suriv 2.01 Suriv 4.02 Tchantches
April 1st—April 30th	Akuku (Wilbur 3) Death Dragon
April 1st—June 30th	Month 4-6
April 3rd—December 31st	Italian Boy
April 12th	ARCV Friends
April 15th	Casino Murphy (Swami)
April 28th	Arale
May 1st—May 4th	1210
May 1st—May 31st	Kthulhu
May 5th	PS-MPC (Cinco de Mayo)
May 13th & May 17th	Arale
May 26th	Find_Me
June 6th	Jerusalem (Sub-Zero B) Psychosis Tiny Virus (Kennedy)
June 12th	Arale June 12th
June 14th	Gremlin

TABLE 48-1 ACTIVATION DATES OF MANY "TRADITIONAL" COMPUTER VIRUSES *(CONTINUED)*

ACTIVATION DATE/DAY	VIRUS NAME
June 16th	June 16th
June 17th—December 31st	Jerusalem (June 17th)
June 26th	DOSHunter
June 28th	Crazy Eddie
July 1st—July 31st	ARCV 330
July 1st—December 31st	Got-You Jerusalem (Jerusalem-PLO) Jerusalem (Mendoza)
July 4th	VCL (Beva 96)
July 13th	July 13th
July 15th	Arale
July 26th	July 26th
August 15th	Casino
August 16th	August 16th
August 22nd	Hare
August 31st	Bomber
September 1st—September 30th	AirCop (AirCop-B) Cascade Sad TenBytes
September 4th	Violator (Violator B1)
September 8th	RIP-699
September 16th	It (Viva Mexico)
September 20th—December 31st	Plastique Plastique-B
September 22nd	Hare
September 22nd—December 31st	4096
October 1st—December 31st	4096 Cascade TenBytes Violator (Violator-C)
October 4th	Violator (Violator B1)
October 12th	Akuku (Columbus) Jerusalem (Anarkia-B)
October 13th—December 31st	Datacrime
October 15th	Dark End
October 23rd	Karin
October 28th	Aragorn
October 30th	Gotcha (Gotcha-Mut4)
October 31st	Halloween Violator (Violator B2)
November, first Tuesday of	Little Brother (LB-349)

TABLE 48-1 ACTIVATION DATES OF MANY "TRADITIONAL" COMPUTER VIRUSES *(CONTINUED)*

ACTIVATION DATE/DAY	VIRUS NAME
November 1st	Maltese Amoeba
November 4th	Violator (Violator B1)
November 11th	Flower
November 12th	Timor
November 17th	November 17th
November 17th—December 31st	November 17th (Nov 17-880)
November 18th	Tiny Virus (Kennedy)
November 22nd	Tiny Virus (Kennedy)
November 24th	PS-MPC (Love Bink)
November 30th	Jerusalem 11-30 Sampo
December 1st—December 31st	1253 Int10
December 1st	Ant
December 4th	Violator (Violator B1)
December 7th	VCL (Pearl Harbor)
December 12th	Arale
December 19th—December 31st	Father Christmas
December 20th—December 25th	ARCV Xmas
December 21st	Poem
December 24th	Icelandic (Icelandic-III)
December 24th—December 31st	Witcode
December 24th—January 1st	Christmas Tree Merry Xmas
December 25th	Black Hawk Japanese Christmas Violator (Violator B3)
December 26th	Find_Me
December 28th	Ash (Ash.546) Spanish April Fools
December 31st	Violator (Violator B2)
After August 1, 1989	Fu Manchu
After June, 1990	Flash
After August, 1990	DataLock
After August 14, 1990	Violator
After November 11, 1990	Fingers
After December 31, 1991	Sicilian Mob
After December 31, 1992	CyberTech OMT
After January 1, 1993	Grunt-1
After December 31, 1993	CyberTech (CyberTech-B)

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES

JANUARY	
1	W97M/Chantal WM/Theatre.A WM/BADBOY.A;B;C WM/CVCK1.B;E WM/JAJA.A Tribute.A;B Acid.A (intended)
2	Flip WM/Alliance.A WM/Helper.C;D;E Grass.A:De
3	WM/HELPER.F;G;H WM/Eraser.A:Tw
4	WM/Eraser.A:Tw WM/Helper.C;D;E
5	WM/HELPER.F;G;H WM/Eraser.A:Tw W97M/Jackal.A AOS.A
6	WM/Eraser.A:Tw WM/KOMPU.A WM/Helper.C;D;E
7	WM/ERASER.H WM/Eraser.A:Tw WM/Alliance.A
8	WM/Eraser.A:Tw WM/KOMPU.A
9	WM/Eraser.A:Tw TRASHER.D W97M/Jackal.A Acid.A (intended)
10	WM/Eraser.A:Tw WM/Helper.A;B AOS.A
11	WM/Eraser.A:Tw WM/Alliance.A WM/MERCY.B WM/CVCK1.A WM/JUNKFACE.A;B
12	WM/Eraser.A:Tw WM/Envader.A (Intended) WM/Alliance.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

JANUARY	
13	WM/FRIDAY.D WM/Goldsecret.B:Int WM/Eraser.A:Tw WM/Envader.A (Intended) WM/BOOM.A;B Twno.A WM/BADBOY.A;B;C WM/FRIDAY.A WM/CVCK1.B;E
14	WM/PHARDERA.C ;D (INTENDED) W97M/Class.B W97M/Class.D WM/Eraser.A:Tw
15	WM/Eraser.A:Tw WM/Theatre.A WM/GANGSTERZ.A W97M/Jackal.A Tribute.A;B AOS.A
16	WM/ERASER.H WM/Eraser.A:Tw WM/Concept.F;G;J Tribute.A;B
17	WM/Eraser.A:Tw W97M/Jackal.A Acid.A (intended)
18	WM/Eraser.A:Tw
19	WM/Eraser.A:Tw Tribute.A;B
20	WM/Eraser.A:Tw WM/MDMA.C;D;H Gurre.A AOS.A
21	WM/MDMA.C;D;H
22	WM/MDMA.C;D;H
23	WM/MDMA.C;D;H TRASHER.D W97M/Melissa.o
24	WM/MDMA.C;D;H WM/CONCEPT.L;M
25	WM/MDMA.C;D;H W97M/Jackal.A Acid.A (intended) AOS.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

JANUARY	
26	WM/TAMAGO.A W32/CIH.Spacefiller WM/MDMA.C;D;H
27	WM/HELPER.F;G;H WM/MDMA.C;D;H W97M/Jackal.A
28	WM/Eraser.B;C:Tw WM/MDMA.C;D;H
29	WM/MDMA.C;D;H
30	WM/MDMA.C;D;H FHD.A W97M/Jackal.A Tribute.A;B AOS.A
31	WM/PHARDERA.C ;D (INTENDED) W97M/Caligula.a WM/MDMA.C;D;H FHD.A WM/CVCK1.A Tribute.A;B
FEBRUARY	
1	WM/Theatre.A WM/BADBOY.A;B;C WM/CVCK1.B;E Tribute.A;B Acid.A (intended)
2	Flip WM/Alliance.A WM/Helper.C;D;E
3	WM/HELPER.F;G;H WM/Eraser.A:Tw
4	WM/Eraser.A:Tw WM/Helper.C;D;E Habir.A
5	WM/HELPER.F;G;H WM/Eraser.A:Tw W97M/Jackal.A AOS.A
6	WM/Eraser.A:Tw WM/KOMPU.A WM/Helper.C;D;E
7	WM/ERASER.H WM/Eraser.A:Tw WM/Alliance.A
8	WM/Eraser.A:Tw WM/KOMPU.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

FEBRUARY	
9	WM/Eraser.A:Tw TRASHER.D W97M/Jackal.A Acid.A (intended) AOS.A
10	WM/Eraser.A:Tw WM/Helper.A;B Grass.A:De AOS.A
11	WM/Eraser.A:Tw WM/Alliance.A WM/MERCY.B WM/CVCK1.A WM/JUNKFACE.A;B
12	WM/Eraser.A:Tw WM/Envader.A (Intended) WM/Alliance.A
13	WM/FRIDAY.D WM/Goldsecret.B:Int WM/Eraser.A:Tw WM/Envader.A (Intended) WM/BOOM.A;B Twno.A WM/BADBOY.A;B;C WM/FRIDAY.A WM/CVCK1.B;E
14	WM/PHARDERA.C ;D (INTENDED) W97M/Class.B W97M/Class.D WM/Eraser.A:Tw
15	WM/Eraser.A:Tw WM/Theatre.A W97M/Jackal.A Tribute.A;B AOS.A
16	WM/ERASER.H WM/Eraser.A:Tw WM/Concept.F;G;J Tribute.A;B
17	WM/Eraser.A:Tw W97M/Jackal.A Acid.A (intended)
18	WM/Eraser.A:Tw
19	WM/Eraser.A:Tw Tribute.A;B

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

FEBRUARY	
20	WM/Eraser.A:Tw WM/MDMA.C;D;H Gurre.A AOS.A
21	WM/MDMA.C;D;H
22	Peter_II W97M/Marker.p WM/MDMA.C;D;H W97M/Stun.a
23	WM/MDMA.C;D;H TRASHER.D W97M/Melissa.o
24	WM/MDMA.C;D;H WM/CONCEPT.L;M
25	WM/MDMA.C;D;H W97M/Jackal.A Acid.A (intended) AOS.A
26	WM/TAMAGO.A W32/CIH.Spacefiller WM/MDMA.C;D;H
27	WM/HELPER.F;G;H WM/MDMA.C;D;H W97M/Jackal.A
28	WM/Eraser.B;C:Tw WM/MDMA.C;D;H
29	WM/MDMA.C;D;H
MARCH	
1	WM/Theatre.A WM/BADBOY.A;B;C WM/CVCK1.B;E WM/ANDRY.A Tribute.A;B Acid.A (intended)
2	Flip WM/Alliance.A WM/Helper.C;D;E
3	WM/HELPER.F;G;H WM/Eraser.A:Tw
4	WM/Eraser.A:Tw WM/Helper.C;D;E
5	WM/HELPER.F;G;H WM/Eraser.A:Tw W97M/Jackal.A AOS.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

MARCH	
6	WM/Eraser.A:Tw WM/KOMPU.A WM/Helper.C;D;E
7	WM/ERASER.H WM/Eraser.A:Tw WM/Alliance.A
8	WM/Eraser.A:Tw WM/KOMPU.A
9	WM/Eraser.A:Tw TRASHER.D WM/HELLGATE.A W97M/Jackal.A Acid.A (intended)
10	WM/Eraser.A:Tw WM/Helper.A;B Grass.A:De AOS.A
11	WM/Eraser.A:Tw WM/Alliance.A WM/MERCY.B WM/CVCK1.A WM/JUNKFACE.A;B
12	WM/Eraser.A:Tw WM/Envader.A (Intended) WM/Alliance.A
13	WM/FRIDAY.D WM/Goldsecret.B:Int WM/Eraser.A:Tw WM/Envader.A (Intended) WM/BOOM.A;B Twno.A WM/BADBOY.A;B;C WM/FRIDAY.A WM/CVCK1.B;E
14	WM/PHARDERA.C ;D (INTENDED) W97M/Class.B W97M/Class.D WM/Eraser.A:Tw
15	WM/Eraser.A:Tw WM/Theatre.A W97M/Jackal.A Tribute.A;B AOS.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

MARCH	
16	WM/ERASER.H WM/Eraser.A:Tw WM/Concept.F;G;J Tribute.A;B
17	WM/Eraser.A:Tw W97M/Jackal.A Acid.A (intended)
18	WM/Eraser.A:Tw
19	WM/Eraser.A:Tw Tribute.A;B
20	WM/Eraser.A:Tw WM/MDMA.C;D;H Gurre.A AOS.A
21	WM/MDMA.C;D;H
22	WM/MDMA.C;D;H
23	WM/MDMA.C;D;H TRASHER.D W97M/Melissa.o
24	WM/MDMA.C;D;H WM/CONCEPT.L;M
25	WM/MDMA.C;D;H W97M/Jackal.A Acid.A (intended) AOS.A
26	WM/TAMAGO.A W32/CIH.Spacefiller WM/MDMA.C;D;H
27	WM/HELPER.F;G;H WM/MDMA.C;D;H W97M/Jackal.A
28	WM/Eraser.B;C:Tw WM/MDMA.C;D;H
29	WM/MDMA.C;D;H
30	WM/MDMA.C;D;H FHD.A W97M/Jackal.A Tribute.A;B AOS.A
31	WM/PHARDERA.C ;D (INTENDED) W97M/Caligula.a WM/ZMB.A:DE WM/MDMA.C;D;H FHD.A WM/CVCK1.A Tribute.A;B

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

APRIL	
1	WM/Theatre.A WM/BADBOY.A;B;C WM/CEEFOUR.A WM/CVCK1.B;E WM/CEEFOUR.B Tribute.A;B Acid.A (intended)
2	Flip WM/Alliance.A WM/Helper.C;D;E
3	WM/HELPER.F;G;H WM/Eraser.A:Tw
4	WM/TWOLINES.A;A1 WM/Eraser.A:Tw WM/Helper.C;D;E
5	WM/HELPER.F;G;H WM/Eraser.A:Tw WM/NUCLEAR.A;B W97M/Jackal.A AOS.A
6	WM/Eraser.A:Tw WM/KOMPU.A WM/Helper.C;D;E
7	WM/ERASER.H WM/Eraser.A:Tw WM/Alliance.A
8	WM/Eraser.A:Tw WM/KOMPU.A
9	WM/Eraser.A:Tw TRASHER.D W97M/Jackal.A Acid.A (intended)
10	WM/Eraser.A:Tw WM/Helper.A;B AOS.A
11	WM/Eraser.A:Tw WM/Alliance.A WM/MERCY.B WM/CVCK1.A WM/JUNKFACE.A;B
12	WM/Eraser.A:Tw WM/Envader.A (Intended) WM/Alliance.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

APRIL	
13	WM/FRIDAY.D WM/Goldsecret.B:Int WM/Eraser.A:Tw WM/Envader.A (Intended) WM/BOOM.A;B Twno.A WM/BADBOY.A;B;C WM/FRIDAY.A WM/MAGNUM.A WM/CVCK1.B;E
14	WM/PHARDERA.C ;D (INTENDED) W97M/Class.B W97M/Class.D WM/Eraser.A:Tw
15	WM/Eraser.A:Tw WM/Theatre.A W97M/Jackal.A Tribute.A;B AOS.A
16	WM/ERASER.H WM/Eraser.A:Tw WM/Concept.F;G;J Tribute.A;B
17	WM/Eraser.A:Tw W97M/Jackal.A Acid.A (intended)
18	WM/Eraser.A:Tw
19	WM/Eraser.A:Tw Tribute.A;B
20	WM/Eraser.A:Tw WM/MDMA.C;D;H Gurre.A AOS.A
21	WM/MDMA.C;D;H
22	WM/MDMA.C;D;H
23	WM/MDMA.C;D;H TRASHER.D W97M/Melissa.o
24	WM/MDMA.C;D;H WM/CONCEPT.L;M
25	WM/MDMA.C;D;H W97M/Jackal.A Acid.A (intended) AOS.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

APRIL	
26	WM/TAMAGO.A W32/CIH.Spacefiller WM/MDMA.C;D;H
27	WM/HELPER.F;G;H WM/MDMA.C;D;H W97M/Jackal.A
28	WM/Eraser.B;C:Tw WM/MDMA.C;D;H
29	WM/MDMA.C;D;H
30	WM/MDMA.C;D;H FHD.A W97M/Jackal.A Tribute.A;B AOS.A
MAY	
1	WM/Theatre.A WM/BADBOY.A;B;C WM/CVCK1.B;E Tribute.A;B Acid.A (intended)
2	Flip WM/Alliance.A WM/Helper.C;D;E
3	WM/HELPER.F;G;H WM/Eraser.A:Tw
4	WM/Eraser.A:Tw WM/Helper.C;D;E
5	WM/HELPER.F;G;H WM/Eraser.A:Tw W97M/Jackal.A AOS.A
6	WM/Eraser.A:Tw WM/KOMPU.A WM/Helper.C;D;E
7	WM/ERASER.H WM/Eraser.A:Tw WM/Alliance.A
8	WM/Eraser.A:Tw WM/KOMPU.A
9	WM/Eraser.A:Tw TRASHER.D W97M/Jackal.A Acid.A (intended)

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

MAY	
10	WM/Eraser.A:Tw WM/Helper.A;B Grass.A:De AOS.A
11	WM/Eraser.A:Tw WM/Alliance.A WM/MERCY.B WM/CVCK1.A WM/JUNKFACE.A;B
12	WM/Eraser.A:Tw WM/Envader.A (Intended) WM/Alliance.A
13	WM/FRIDAY.D WM/Goldsecret.B:Int WM/Eraser.A:Tw WM/Envader.A (Intended) WM/BOOM.A;B Twno.A WM/BADBOY.A;B;C WM/FRIDAY.A WM/CVCK1.B;E
14	WM/PHARDERA.C ;D (INTENDED) W97M/Class.B W97M/Class.D WM/Eraser.A:Tw
15	WM/Eraser.A:Tw WM/Theatre.A W97M/Jackal.A Tribute.A;B AOS.A
16	WM/ERASER.H WM/Eraser.A:Tw WM/Concept.F;G;J Tribute.A;B
17	WM/Eraser.A:Tw W97M/Jackal.A Acid.A (intended)
18	WM/Eraser.A:Tw
19	WM/Eraser.A:Tw Tribute.A;B
20	WM/Eraser.A:Tw WM/MDMA.C;D;H Gurre.A AOS.A
21	WM/MDMA.C;D;H

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

MAY	
22	WM/MDMA.C;D;H
23	WM/MDMA.C;D;H TRASHER.D W97M/Melissa.o
24	WM/MDMA.C;D;H WM/CONCEPT.L;M
25	WM/MDMA.C;D;H W97M/Jackal.A Acid.A (intended) AOS.A
26	WM/TAMAGO.A W32/CIH.Spacefiller WM/MDMA.C;D;H
27	WM/HELPER.F;G;H WM/MDMA.C;D;H W97M/Jackal.A
28	WM/Eraser.B;C:Tw WM/MDMA.C;D;H
29	WM/MDMA.C;D;H
30	WM/MDMA.C;D;H FHD.A W97M/Jackal.A Tribute.A;B AOS.A
31	WM/PHARDERA.C ;D (INTENDED) W97M/Caligula.a WM/MDMA.C;D;H FHD.A WM/CVCK1.A Tribute.A;B
JUNE	
1	WM/HARK.A WM/Theatre.A WM/BADBOY.A;B;C WM/CVCK1.B;E Tribute.A;B Acid.A (intended)
2	Flip WM/Alliance.A WM/Helper.C;D;E
3	WM/HELPER.F;G;H WM/Eraser.A:Tw
4	WM/Eraser.A:Tw WM/Helper.C;D;E

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

JUNE	
5	WM/HELPER.F;G;H WM/Eraser.A:Tw W97M/Jackal.A AOS.A
6	WM/Eraser.A:Tw WM/KOMPU.A WM/Helper.C;D;E
7	WM/ERASER.H WM/Eraser.A:Tw WM/Alliance.A
8	WM/Eraser.A:Tw WM/KOMPU.A
9	WM/Eraser.A:Tw TRASHER.D W97M/Jackal.A Acid.A (intended)
10	WM/Eraser.A:Tw WM/Helper.A;B AOS.A
11	WM/Eraser.A:Tw WM/Alliance.A WM/MERCY.B WM/CVCK1.A WM/JUNKFACE.A;B
12	WM/Eraser.A:Tw WM/Envader.A (Intended) WM/Alliance.A
13	WM/FRIDAY.D WM/Goldsecret.B:Int WM/Eraser.A:Tw WM/Envader.A (Intended) WM/BOOM.A;B Twno.A WM/BADBOY.A;B;C WM/FRIDAY.A WM/CVCK1.B;E
14	WM/PHARDERA.C ;D (INTENDED) W97M/Class.B W97M/Class.D WM/Eraser.A:Tw
15	WM/Eraser.A:Tw WM/Theatre.A W97M/Jackal.A Tribute.A;B AOS.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

JUNE	
16	WM/ERASER.H WM/Eraser.A:Tw WM/Concept.F;G;J Tribute.A;B
17	WM/Eraser.A:Tw W97M/Jackal.A Acid.A (intended)
18	WM/Eraser.A:Tw
19	WM/Eraser.A:Tw Tribute.A;B
20	WM/Eraser.A:Tw WM/MDMA.C;D;H Gurre.A AOS.A
21	Werewolf.1168 WM/MDMA.C;D;H
22	WM/MDMA.C;D;H
23	WM/MDMA.C;D;H TRASHER.D W97M/Melissa.o
24	WM/MDMA.C;D;H WM/CONCEPT.L;M
25	WM/MDMA.C;D;H W97M/Jackal.A Acid.A (intended) AOS.A
26	WM/TAMAGO.A W32/CIH.Spacefiller WM/MDMA.C;D;H
27	WM/HELPER.F;G;H WM/MDMA.C;D;H W97M/Jackal.A
28	WM/Eraser.B;C:Tw WM/MDMA.C;D;H
29	WM/MDMA.C;D;H
30	WM/MDMA.C;D;H FHD.A W97M/Jackal.A Tribute.A;B AOS.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

JULY	
1	WM/BALROG.A WM/Theatre.A WM/BADBOY.A;B;C WM/CVCK1.B;E Tribute.A;B Acid.A (intended)
2	Flip WM/Alliance.A WM/Helper.C;D;E
3	WM/HELPER.F;G;H WM/Eraser.A:Tw
4	WM/Eraser.A:Tw WM/Helper.C;D;E
5	WM/HELPER.F;G;H WM/Eraser.A:Tw W97M/Jackal.A AOS.A
6	WM/Eraser.A:Tw WM/KOMPU.A WM/Helper.C;D;E
7	WM/ERASER.H WM/Eraser.A:Tw WM/Alliance.A
8	WM/Eraser.A:Tw WM/KOMPU.A
9	WM/ANGEL.A WM/Eraser.A:Tw TRASHER.D W97M/Jackal.A Acid.A (intended)
10	WM/Eraser.A:Tw WM/Helper.A;B AOS.A
11	WM/Eraser.A:Tw WM/Alliance.A WM/MERCY.B WM/CVCK1.A WM/JUNKFACE.A;B
12	WM/Eraser.A:Tw WM/Envader.A (Intended) WM/Alliance.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

JULY	
13	WM/FRIDAY.D WM/Goldsecret.B:Int WM/Eraser.A:Tw WM/Envader.A (Intended) WM/BOOM.A;B Twno.A WM/BADBOY.A;B;C WM/FRIDAY.A WM/CVCK1.B;E
14	WM/PHARDERA.C ;D (INTENDED) W97M/Class.B W97M/Class.D WM/Eraser.A:Tw
15	WM/Eraser.A:Tw WM/Theatre.A W97M/Jackal.A Tribute.A;B AOS.A
16	WM/ERASER.H WM/Eraser.A:Tw WM/Concept.F;G;J Tribute.A;B
17	WM/Eraser.A:Tw W97M/Jackal.A Acid.A (intended)
18	WM/Eraser.A:Tw
19	WM/Eraser.A:Tw Tribute.A;B
20	WM/Eraser.A:Tw WM/MDMA.C;D;H Gurre.A AOS.A
21	WM/MDMA.C;D;H
22	WM/MDMA.C;D;H W97M/Stun.a
23	WM/MDMA.C;D;H TRASHER.D W97M/Melissa.o
24	WM/MDMA.C;D;H WM/CONCEPT.L;M
25	WM/MDMA.C;D;H W97M/Jackal.A Acid.A (intended) AOS.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

JULY	
26	WM/TAMAGO.A W32/CIH.Spacefiller WM/MDMA.C;D;H
27	WM/HELPER.F;G;H WM/MDMA.C;D;H W97M/Jackal.A
28	WM/Eraser.B;C:Tw WM/MDMA.C;D;H
29	WM/MDMA.C;D;H
30	WM/MDMA.C;D;H FHD.A W97M/Jackal.A Tribute.A;B AOS.A
31	WM/PHARDERA.C ;D (INTENDED) W97M/Caligula.a WM/MDMA.C;D;H FHD.A WM/CVCK1.A Tribute.A;B
AUGUST	
1	WM/Theatre.A WM/BADBOY.A;B;C WM/CVCK1.B;E Tribute.A;B Acid.A (intended)
2	Flip WM/Alliance.A WM/Helper.C;D;E
3	WM/HELPER.F;G;H WM/Eraser.A:Tw
4	WM/Eraser.A:Tw WM/Helper.C;D;E
5	WM/HELPER.F;G;H WM/Eraser.A:Tw W97M/Jackal.A AOS.A
6	WM/Eraser.A:Tw WM/KOMPU.A WM/Helper.C;D;E
7	WM/ERASER.H WM/Eraser.A:Tw WM/Alliance.A
8	WM/Eraser.A:Tw WM/KOMPU.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

AUGUST	
9	WM/Eraser.A:Tw TRASHER.D W97M/Jackal.A Acid.A (intended)
10	WM/Eraser.A:Tw WM/Helper.A;B AOS.A
11	WM/Eraser.A:Tw WM/Alliance.A WM/MERCY.B WM/CVCK1.A WM/JUNKFACE.A;B
12	WM/Eraser.A:Tw WM/Envader.A (Intended) WM/Alliance.A
13	WM/FRIDAY.D WM/Goldsecret.B:Int WM/Eraser.A:Tw WM/Envader.A (Intended) WM/BOOM.A;B Twno.A WM/BADBOY.A;B;C WM/FRIDAY.A WM/CVCK1.B;E
14	WM/PHARDERA.C ;D (INTENDED) W97M/Class.B W97M/Class.D WM/Eraser.A:Tw
15	WM/Eraser.A:Tw WM/Theatre.A W97M/Jackal.A Tribute.A;B AOS.A
16	WM/ERASER.H WM/Eraser.A:Tw WM/Concept.F;G;J Tribute.A;B
17	WM/Eraser.A:Tw W97M/Jackal.A Acid.A (intended)
18	WM/Eraser.A:Tw
19	WM/Eraser.A:Tw Tribute.A;B
20	WM/Eraser.A:Tw WM/MDMA.C;D;H Gurre.A AOS.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

AUGUST	
21	WM/MDMA.C;D;H
22	WM/MDMA.C;D;H
23	WM/MDMA.C;D;H TRASHER.D W97M/Melissa.o
24	WM/MDMA.C;D;H WM/CONCEPT.L;M
25	WM/MDMA.C;D;H W97M/Jackal.A Acid.A (intended) AOS.A
26	WM/TAMAGO.A W32/CIH.Spacefiller WM/MDMA.C;D;H
27	WM/HELPER.F;G;H WM/MDMA.C;D;H W97M/Jackal.A
28	WM/Eraser.B;C:Tw WM/MDMA.C;D;H
29	WM/MDMA.C;D;H
30	WM/MDMA.C;D;H FHD.A W97M/Jackal.A Tribute.A;B AOS.A
31	WM/PHARDERA.C ;D (INTENDED) W97M/Caligula.a WM/MDMA.C;D;H FHD.A WM/CVCK1.A Tribute.A;B
SEPTEMBER	
1	WM/Theatre.A WM/BADBOY.A;B;C WM/CVCK1.B;E Tribute.A;B Acid.A (intended)
2	Flip WM/Alliance.A WM/Helper.C;D;E
3	WM/HELPER.F;G;H WM/Eraser.A:Tw
4	WM/Eraser.A:Tw WM/Helper.C;D;E

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

SEPTEMBER	
5	WM/HELPER.F;G;H WM/Eraser.A:Tw W97M/Jackal.A AOS.A
6	WM/Eraser.A:Tw WM/KOMPU.A WM/Helper.C;D;E
7	WM/ERASER.H WM/Eraser.A:Tw WM/Alliance.A
8	WM/Eraser.A:Tw WM/KOMPU.A
9	WM/Eraser.A:Tw TRASHER.D W97M/Jackal.A Acid.A (intended)
10	WM/Eraser.A:Tw WM/Helper.A;B AOS.A
11	WM/Eraser.A:Tw WM/Alliance.A WM/MERCY.B WM/CVCK1.A WM/JUNKFACE.A;B
12	WM/Eraser.A:Tw WM/Envader.A (Intended) WM/Alliance.A
13	WM/FRIDAY.D WM/Goldsecret.B:Int WM/Eraser.A:Tw WM/Envader.A (Intended) WM/BOOM.A;B Twno.A WM/BADBOY.A;B;C WM/FRIDAY.A WM/CVCK1.B;E
14	WM/PHARDERA.C ;D (INTENDED) W97M/Class.B W97M/Class.D WM/Eraser.A:Tw
15	WM/Eraser.A:Tw WM/Theatre.A W97M/Jackal.A Tribute.A;B AOS.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

SEPTEMBER	
16	WM/ERASER.H WM/Eraser.A:Tw WM/Concept.F;G;J Tribute.A;B
17	WM/Eraser.A:Tw W97M/Jackal.A Acid.A (intended)
18	WM/Eraser.A:Tw
19	WM/Eraser.A:Tw Tribute.A;B
20	WM/Eraser.A:Tw WM/MDMA.C;D;H Gurre.A AOS.A
21	WM/MDMA.C;D;H
22	WM/MDMA.C;D;H
23	WM/MDMA.C;D;H TRASHER.D W97M/Melissa.o
24	WM/MDMA.C;D;H WM/CONCEPT.L;M
25	WM/MDMA.C;D;H W97M/Jackal.A Acid.A (intended) AOS.A
26	WM/TAMAGO.A W32/CIH.Spacefiller WM/MDMA.C;D;H
27	WM/HELPER.F;G;H WM/MDMA.C;D;H W97M/Jackal.A
28	WM/Eraser.B;C:Tw WM/MDMA.C;D;H
29	WM/MDMA.C;D;H
30	WM/MDMA.C;D;H FHD.A W97M/Jackal.A Tribute.A;B AOS.A
OCTOBER	
1	WM/Theatre.A WM/BADBOY.A;B;C WM/CVCK1.B;E Tribute.A;B Acid.A (intended)

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

OCTOBER	
2	Flip WM/Alliance.A WM/Helper.C;D;E
3	WM/HELPER.F;G;H WM/Eraser.A:Tw
4	WM/Eraser.A:Tw WM/Helper.C;D;E
5	WM/HELPER.F;G;H WM/Eraser.A:Tw W97M/Jackal.A AOS.A
6	WM/Eraser.A:Tw WM/KOMPU.A WM/Helper.C;D;E
7	WM/ERASER.H WM/Eraser.A:Tw WM/Alliance.A
8	WM/Eraser.A:Tw WM/KOMPU.A
9	WM/Eraser.A:Tw TRASHER.D W97M/Jackal.A Acid.A (intended)
10	WM/Eraser.A:Tw WM/Helper.A;B AOS.A
11	W97M/AntiSocial.e WM/Eraser.A:Tw WM/Alliance.A WM/MERCY.B WM/CVCK1.A WM/JUNKFACE.A;B
12	WM/Eraser.A:Tw WM/Envader.A (Intended) WM/Alliance.A
13	WM/FRIDAY.D WM/Goldsecret.B:Int WM/Eraser.A:Tw WM/Envader.A (Intended) WM/BOOM.A;B Twno.A WM/BADBOY.A;B;C WM/FRIDAY.A WM/CVCK1.B;E

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

OCTOBER	
14	WM/PHARDERA.C ;D (INTENDED) W97M/Class.B W97M/Class.D WM/Eraser.A:Tw
15	WM/Eraser.A:Tw WM/Theatre.A W97M/Jackal.A Tribute.A;B AOS.A Habir.A
16	WM/ERASER.H WM/Eraser.A:Tw WM/Concept.F;G;J Tribute.A;B
17	WM/Eraser.A:Tw W97M/Jackal.A Acid.A (intended)
18	WM/Eraser.A:Tw
19	WM/Eraser.A:Tw Tribute.A;B
20	WM/Eraser.A:Tw WM/MDMA.C;D;H Gurre.A AOS.A
21	WM/MDMA.C;D;H
22	WM/MDMA.C;D;H
23	WM/Niknat.A WM/MDMA.C;D;H TRASHER.D W97M/Melissa.o WM/ANGUS.A WM/ANGUS.A
24	WM/MDMA.C;D;H WM/CONCEPT.L;M WM/ANGUS.A
25	WM/MDMA.C;D;H W97M/Jackal.A Acid.A (intended) AOS.A
26	WM/TAMAGO.A W32/CIH.Spacefiller WM/MDMA.C;D;H
27	WM/HELPER.F;G;H WM/MDMA.C;D;H W97M/Jackal.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

OCTOBER	
28	WM/Eraser.B;C:Tw WM/MDMA.C;D;H
29	WM/MDMA.C;D;H
30	WM/MDMA.C;D;H FHD.A W97M/Jackal.A Tribute.A;B AOS.A
31	WM/PHARDERA.C ;D (INTENDED) W97M/Caligula.a WM/MDMA.C;D;H FHD.A WM/CVCK1.A Tribute.A;B
NOVEMBER	
1	WM/Theatre.A WM/BADBOY.A;B;C WM/CVCK1.B;E Tribute.A;B Acid.A (intended)
2	Flip WM/Alliance.A WM/Helper.C;D;E
3	WM/HELPER.F;G;H WM/Eraser.A:Tw
4	WM/Eraser.A:Tw WM/Helper.C;D;E
5	WM/HELPER.F;G;H WM/Eraser.A:Tw W97M/Jackal.A AOS.A
6	WM/Eraser.A:Tw WM/KOMPU.A WM/Helper.C;D;E
7	WM/ERASER.H WM/Eraser.A:Tw WM/Alliance.A
8	WM/Eraser.A:Tw WM/KOMPU.A
9	Delta.1163 WM/Eraser.A:Tw TRASHER.D W97M/Jackal.A Acid.A (intended)

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

NOVEMBER	
10	WM/Eraser.A:Tw WM/Helper.A;B AOS.A
11	WM/ANGEL.A WM/Eraser.A:Tw WM/Alliance.A WM/MERCY.B WM/CVCK1.A WM/JUNKFACE.A;B
12	WM/Eraser.A:Tw WM/Envader.A (Intended) WM/Alliance.A
13	Deadwin.1228 WM/FRIDAY.D WM/Goldsecret.B:Int WM/Eraser.A:Tw WM/Envader.A (Intended) WM/BOOM.A;B Twno.A WM/BADBOY.A;B;C WM/FRIDAY.A WM/CVCK1.B;E
14	WM/PHARDERA.C ;D (INTENDED) W97M/Class.B W97M/Class.D WM/Eraser.A:Tw
15	WM/Eraser.A:Tw WM/Theatre.A W97M/Jackal.A Tribute.A;B AOS.A
16	WM/ERASER.H WM/Eraser.A:Tw WM/Concept.F;G;J Tribute.A;B
17	WM/Eraser.A:Tw W97M/Jackal.A Acid.A (intended)
18	WM/Eraser.A:Tw
19	WM/Eraser.A:Tw Tribute.A;B
20	WM/Eraser.A:Tw WM/MDMA.C;D;H Gurre.A AOS.A

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

NOVEMBER	
21	WM/MDMA.C;D;H
22	WM/MDMA.C;D;H
23	WM/MDMA.C;D;H TRASHER.D W97M/Melissa.o
24	WM/MDMA.C;D;H WM/CONCEPT.L;M
25	WM/MDMA.C;D;H W97M/Jackal.A Acid.A (intended) AOS.A
26	WM/TAMAGO.A W32/CIH.Spacefiller WM/MDMA.C;D;H
27	WM/HELPER.F;G;H WM/MDMA.C;D;H W97M/Jackal.A
28	WM/Eraser.B;C:Tw WM/MDMA.C;D;H
29	WM/MDMA.C;D;H
30	WM/MDMA.C;D;H FHD.A W97M/Jackal.A Tribute.A;B AOS.A
DECEMBER	
1	WM/Theatre.A WM/BADBOY.A;B;C WM/CVCK1.B;E Tribute.A;B Acid.A (intended)
2	Flip WM/Alliance.A WM/Helper.C;D;E
3	WM/HELPER.F;G;H WM/Eraser.A:Tw
4	WM/Eraser.A:Tw WM/Helper.C;D;E
5	WM/HELPER.F;G;H WM/Eraser.A:Tw WM/ATOM.H W97M/Jackal.A AOS.A
6	WM/Eraser.A:Tw WM/KOMPU.A WM/Helper.C;D;E

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

DECEMBER	
7	WM/ERASER.H WM/Eraser.A:Tw WM/Alliance.A
8	WM/Eraser.A:Tw WM/KOMPU.A
9	WM/Eraser.A:Tw TRASHER.D W97M/Jackal.A Acid.A (intended)
10	WM/Eraser.A:Tw WM/Helper.A;B AOS.A
11	WM/Eraser.A:Tw WM/Alliance.A WM/MERCY.B WM/CVCK1.A WM/JUNKFACE.A;B
12	WM/Eraser.A:Tw WM/Envader.A (Intended) WM/Alliance.A
13	W97M/Nottice.A WM/FRIDAY.D WM/Goldsecret.B:Int WM/Eraser.A:Tw WM/Envader.A (Intended) WM/Atom.A;E;I;J WM/BOOM.A;B Twno.A WM/BADBOY.A;B;C WM/ATOM.G WM/FRIDAY.A WM/CVCK1.B;E
14	WM/PHARDERA.C ;D (INTENDED) W97M/Class.B W97M/Class.D WM/Eraser.A:Tw
15	WM/Eraser.A:Tw WM/Theatre.A W97M/Jackal.A Tribute.A;B AOS.A
16	WM/ERASER.H WM/Eraser.A:Tw WM/Concept.F;G;J Tribute.A;B

TABLE 48-2 ACTIVATION DATES OF MANY CURRENT COMPUTER VIRUSES *(CONTINUED)*

DECEMBER	
17	WM/Eraser.A:Tw W97M/Jackal.A Acid.A (intended)
18	WM/Eraser.A:Tw
19	WM/Eraser.A:Tw Tribute.A;B
20	WM/Eraser.A:Tw WM/MDMA.C;D;H Gurre.A AOS.A
21	WM/MDMA.C;D;H
22	WM/MDMA.C;D;H
23	W97M/Class.AZ WM/MDMA.C;D;H TRASHER.D W97M/Melissa.o
24	W97M/Class.AZ WM/MDMA.C;D;H WM/CONCEPT.L;M
25	W97M/Class.AZ WM/MDMA.C;D;H W97M/Jackal.A Acid.A (intended) AOS.A
26	WM/TAMAGO.A W32/CIH.Spacefiller WM/MDMA.C;D;H
27	WM/HELPER.F;G;H WM/MDMA.C;D;H W97M/Jackal.A
28	WM/Eraser.B;C:Tw WM/MDMA.C;D;H
29	WM/MDMA.C;D;H
30	WM/MDMA.C;D;H FHD.A W97M/Jackal.A Tribute.A;B AOS.A
31	WM/PHARDERA.C ;D (INTENDED) W97M/Caligula.a WM/MDMA.C;D;H FHD.A WM/CVCK1.A Tribute.A;B

REPLICATORS

The purpose of a *replicator* (also called a *rabbit*) is to drain system resources. It accomplishes this function by cloning copies of itself. Each clone copy is launched by the parent that created it. Before long, the multitude of copies on disk and in memory soaks up so many resources that the system can no longer function In effect, the system is crippled until the copies are removed and the replicating virus is eliminated. This type of behavior is particularly effective at shutting down large, multi-user systems or networks. Since the virulent code is self-replicating, it is easy to spot with antivirus tools.

WORMS

Unlike most other types of virulent code, the *worm* travels through a network computer system. The worm travels from computer to computer—usually without doing any real damage. Worms rarely replicate, except in cases where it is absolutely necessary to continue traveling through the system, and delete all traces of their presence. A worm is another typical network presence used to seek out and selectively alter or destroy a limited number of files or programs. For example, a worm can be used to enter a network and alter or erase passwords. Since worms can be tailored for specific jobs, they are often difficult to spot unless the worm is known.

VIRUSES

The most recognized and dynamic of the rogue software is the *virus*. A virus modifies other programs to include executable virulent code—in some cases, the virulent code mutates and changes as it is copied. Expertly engineered viruses do not change the infected file date, time stamps, size, attributes, or checksums. As a result, viruses can be extremely difficult to detect and even harder to erase—and the task becomes even more difficult as viruses become increasingly powerful and sophisticated. With today's "high overhead" operating systems, such as Windows 95/98 Windows NT, viruses can usually hide and replicate quite easily in any of the numerous DLL files, VXD files, or other modules normally in operation. Given their predilection toward stealth and replication, viruses tend to linger in systems to spread themselves between hard drives and floppy disks and network connections, where they disrupt data, cause system errors, and generally degrade system performance. Eventually, a typical virus will self-destruct, usually taking the hard drive files with it.

Types of Viruses

As you might have suspected, all virulent code is not created equal. Viruses are as varied as legitimate application software—each technique provides the virus author with an array of advantages and disadvantages. Some viral techniques are preferred because they are more difficult to detect and remove, but they require extra resources to develop. Other viral techniques are easier to develop, but lack the stealth and sophistication that more powerful viruses demand. Still other viral techniques stand a better chance of infecting multiple systems. This part of the chapter explains the major infection modes used by modern viruses.

Boot-Sector Viruses

Early PCs loaded their operating systems from floppy disks. Virus authors quickly discovered that they could substitute their own program for the executable code present on the boot sector of every floppy disk formatted with DOS—whether or not it included system files. Unsuspecting users thus loaded the virus into memory every time they started their computers with an infected disk. Once in memory, a virus can copy itself to boot sectors on other floppy or hard disks. For example, those who unintentionally loaded

"Brain" from an infected floppy found themselves reading an "advertisement" for a computer consulting company in Pakistan. With that advertisement, "Brain" pioneered another characteristic feature of modern viruses—the payload. The payload is the prank or malicious behavior that (if triggered) causes effects that range from annoying messages to data destruction. The payload is the virus characteristic that draws the most attention. Many virus authors now write their viruses specifically to deliver their payloads to as many computers as possible.

For a time, sophisticated descendants of this first boot-sector virus represented the most serious virus threat to computer users. Variants of boot sector viruses also infect the Master Boot Record (MBR), which stores partition information. Nearly every step in the boot process (from reading the MBR to loading the operating system) is vulnerable to viral sabotage. Some of the most tenacious and destructive viruses still include the ability to infect your computer's boot sector or MBR. Loading a boot sector virus at boot time can give a virus a chance to do its work before your antivirus software has a chance to run. Some antivirus tools anticipate this possibility by allowing you to create an "emergency disk" that you can use to boot your computer and remove infections.

Boot sector and MBR viruses have a particular weakness—they must spread by means of floppy disks or other removable media, hiding in that first track of disk space. As fewer users exchange floppy disks, and software distribution relies on other media (such as CDs and Internet downloads), other virus types have recently eclipsed the boot sector threat. However, the popularity of large-capacity media "disks," such as Iomega Zip disks and Jaz disks, could cause a resurgence of this virus type.

File-Infector Viruses

At about the same time as the authors of the "Brain" virus found vulnerabilities in the DOS boot sector, other virus writers found out how to use existing software to help replicate their creations. An early example of this type of virus showed up in computers at Lehigh University in Pennsylvania. The virus infected part of the DOS command interpreter (COMMAND.COM), which it used to load itself into memory—once there, it spreads to other uninfected COMMAND.COM files each time a user enters any standard DOS command that involves disk access. Early iterations of this virus limited its spread to floppy disks that contain a full operating system.

Later viruses quickly overcame this limitation—sometimes with fairly clever programming. For example, virus writers might have their virus add its code to the beginning of an executable file. When users start a program, the virus code executes immediately, then transfers control back to the legitimate software (which runs as though nothing unusual has happened). Once it activates, the virus "hooks" or "traps" requests that legitimate software makes to the operating system and substitutes its own responses. Particularly clever viruses can even subvert attempts to clear them from memory by trapping the CTRL+ALT+DEL keyboard sequence for a warm reboot, then faking a restart. Sometimes the only outward indication that anything on your system is amiss (before any payload "detonates") might be a small change in the file size of infected legitimate software.

Stealth, Mutating, Encrypted, and Polymorphic Viruses

As unobtrusive as they might be, changes in file size and other scant evidence of a virus infection usually gives most antivirus software enough of a clue to locate and remove the offending code. One of virus writers' principal challenges is to find ways to hide their handiwork. The earliest disguises were a mixture of innovative programming and obvious giveaways. For instance, the "Brain" virus redirected requests to see a disk's boot sector away from the actual location of the infected sector to the new location of the boot files—which the virus had moved. This "stealth" capability enabled this and other viruses to hide from conventional search techniques.

Because viruses needed to avoid continuously reinfecting host systems (doing so would quickly balloon an infected file's size to easily detectable proportions, or would consume enough system resources to point to an obvious culprit), virus authors also needed to tell the viruses to leave certain files alone. They addressed this problem by having the virus write a code "signature" that would flag infected files with the software equivalent of a "do not disturb" sign. Although that kept the virus from giving itself away immediately, it opened the way for antivirus software to use the code "signatures" themselves to find the virus.

In response, virus writers found ways to conceal the code signatures. Some viruses would "mutate," or write different code signatures, with each new infection. Others encrypted most of the code signature or the virus itself—leaving only a couple of bytes to use as a key for decryption. The most sophisticated new viruses employ stealth, mutation, and encryption to appear in an almost undetectable variety of new forms. Finding these *polymorphic* viruses requires software engineers to develop very elaborate programming techniques for antivirus software.

Macro Viruses

By 1995 or so, the virus war had come to something of a standstill. New viruses appeared continuously (prompted in part by the availability of ready-made virus "kits" that enabled even some non-programmers to whip up a new virus in no time). Most existing antivirus software could easily be updated to detect and dispose of the new virus variants—which consisted primarily of minor tweaks to well-known templates.

But 1995 marked the emergence of the *concept* virus, which added a new and surprising twist to virus history. Before concept, most virus researchers thought of data files (the text, spreadsheet, or drawing documents created by software) as *immune* to infection. Viruses, after all, *are* programs and, as such, need to be run the same way executable software does in order to do their damage. On the other hand, data files are simply stored information that you enter when you work with your software.

That distinction melted away when Microsoft began adding "macro" capabilities to Word and Excel—the flagship applications in its Office suite. Using a stripped-down version of its Visual BASIC language, users could create document templates that would automatically format and add other features to documents created with Word and Excel. Virus writers seized the opportunity that this presented to conceal and spread viruses in documents that you, the user, create yourself. The exploding popularity of the Internet and of e-mail software that allows users to attach files to messages ensured that macro viruses would spread very quickly and very widely. Within a year, macro viruses had become the most potent virus threat ever.

JAVA AND ACTIVEX

Programs based on Java and ActiveX come in a variety of forms. Some are special-purpose miniature applications (or "applets") written in Java—a new programming language first developed by Sun Microsystems. Others are developed using ActiveX—a Microsoft technology that programmers can use for similar purposes.

Both Java and ActiveX make extensive use of prewritten software modules (or "objects") that programmers can write themselves or take from existing sources and fashion into the plug-ins, applets, device drivers, and other software needed to power the web. Java objects are called "classes," while ActiveX objects are called "controls." The principal difference between them lies in how they run on the host system. Java applets run in a Java "virtual machine" designed especially to interpret Java programming and translate it into action on the host machine, while ActiveX controls run as native Windows programs that link and pass data between existing Windows software.

The overwhelming majority of these objects are useful (even necessary) parts of any interactive web site. But despite the best efforts of Sun and Microsoft engineers to design security measures into them,

determined programmers can use Java and ActiveX tools to plant harmful objects on web sites, where they can lurk until visitors unwittingly allow them access to vulnerable computer systems. Unlike viruses, harmful Java and ActiveX objects usually don't seek self-replication as their primary goal—the web provides them with plenty of opportunities to spread to target computer systems, while their small size and innocuous nature makes it easy for them to evade detection. In fact, unless you specifically tell your browser software to block them, Java and ActiveX objects automatically download to your system whenever you visit a web site that hosts them.

Instead, harmful objects exist to deliver their equivalent of a virus payload. For example, programmers have written objects that can read data from your hard disk and send it back to the web site you visited. These objects can "hijack" your e-mail account and send out offensive messages in your name or can watch data that passes between your computer and other computers.

EMERGING VIRUS TYPES

Malicious software has even begun intruding into areas once thought completely out of bounds. Users of the mIRC Internet Relay Chat client, for example, have reported encountering viruses constructed from the mIRC scripting language. Script viruses get sent as plain text (which would ordinarily preclude them from getting infected) but older versions of the mIRC client software would interpret the instructions coded into the script and perform unwanted actions on the recipient's computer. Vendors moved quickly to disable this capability in updated versions of the software, but the mIRC incident illustrates the general rule that where a way exists to exploit a software security hole, someone will find it and use it.

Virus Myths

Computer viruses are a real threat, and one that should always be taken seriously, But in most cases, computer viruses are rarely the harbingers of doom and gloom that many novices (and much of the PC media) perceive them to be. Now that you have an idea of the nature of viruses and other rogue software, it's time dispel some persistent myths surrounding viruses:

- *No antivirus software is 100% effective* Although antivirus products are constantly being updated to protect against the latest virus threats, there is no such thing as a "foolproof" virus protection program—new viruses are constantly being designed to bypass them. The best protection is to scan for viruses regularly using a current antivirus tool, and always keep your vital files backed up.
- *A virus cannot hide inside a data file* Data files (such as images) cannot spread a virus on your computer. Only executable program files (and files containing executable macros) can spread viruses. A computer virus *could* infect a data file, but it would be a useless effort—since a data file is not executed, only *loaded*, the virus would not be able run or to replicate itself.

Text and spreadsheet files supporting macros *can* be "infected" with destructive macros. Scan text and spreadsheet files for macro viruses before loading them.

- *Viruses cannot spread to all types of computers* Viruses are limited to a given family of computers. For example, a virus designed to spread on IBM PCs cannot infect an IBM 4300 series mainframe, or infect a Commodore C64, or infect an Apple Macintosh. However, cross-platform software can spread on any system capable of opening and reading the infected file(s). Word macro viruses can spread on any platform that reads Word files.

- *A computer cannot be infected by calling an infected BBS, FTP, or web site* BBS and FTP sites containing infected files *cannot* write information onto your computer under its own direction. Your communications software (or web browser) performs this task. You can transfer an infected file to your computer only if you let your software do it. If an infected file is transferred to your computer, it *cannot* spread until you *execute* the downloaded file. If a file is scanned after being downloaded—and found to be infected—it can be safely deleted before infecting other components of the computer.
- *Compressed file archives can be infected* Although an "archive" file (for instance, a .ZIP file) cannot be infected itself, the executable files contained in the archive *can* be infected. You can decompress the archive without executing any of the files in the archive, then scan the files with an antivirus tool *before* installing the software or running any of the executable files in the archive.
- *A boot sector virus cannot travel in downloaded software* BBS and Internet download sites deal only in program files, and do not pass along copies of disk boot sectors. Since boot sector viruses can spread only by "booting" (or attempting to boot the computer from an infected disk), downloading is generally immune to boot-sector viruses. However, you should still scan all downloaded files before executing them for the first time.
- *Damaged files do not always indicate a virus attack* This is a very common misconception about viruses. Damaged files can be caused by many things (including the result of a power surge, power drop-off, static electricity, magnetic forces, failing hardware components, a bug in another software package, dust, fingerprints, spilled coffee, and so on). Power failures and spilled cups of coffee have destroyed more data than any viruses. Still, you should run your virus checker just to be sure.
- *Backups are still valuable—even with a virus* Suppose a virus is backed up with your files. It could not be a boot-infecting virus because the backup software will not back up the boot sector. If you had a file-infecting virus, you could restore important documents, databases, and your data, without restoring an infected program—or delete the infected programs and restore them specifically from the original installation disks.
- *Read-only files are* not *immune to infection* Some computer users believe you can protect yourself by using the DOS ATTRIB command to set the read-only attribute on program files. However, ATTRIB is software, and what it can do, a virus can easily undo. While this tactic may be marginally successful at halting very old or simple viruses, the ATTRIB command very rarely halts the spread of viruses.
- *Viruses cannot infect write-protected disks* Since viruses can modify read-only files, people tend to believe they can also modify write-protected floppy disks. The disk drive senses a protected disk and refuses to write on it. This protection is controlled by the hardware—not software. You can physically disable a floppy drive's write-protect sensor, but you cannot override it with a software command. Write-protecting your disks are a free and easy means of halting the spread of viruses—especially boot sector viruses.

Protecting the PC

Even with the most comprehensive, accurate, aggressive, up-to-the-minute antivirus package available, antivirus tools alone will not always protect a PC from the ravages of a virus or other rogue software. Trying a suspicious piece of software without testing it first, forgetting to virus scan the system regularly, and even intentional sabotage can render an antivirus tool useless. Before trouble strikes, you can take some pro-active steps to prevent the spread of viruses and to ease your recovery should a virus actually strike:

- *Check for viruses regularly* You would be surprised how many people buy antivirus products only to use them sporadically or leave them sitting unused until it is too late. Remember that antivirus tools are always behind viruses—you need to use your antivirus tools consistently and aggressively in order to catch viruses before they do their damage. If you are regularly trying new shareware or commercial products, you should be sure to check for viruses religiously. Also check for viruses if you routinely swap disks between home and work PCs or a variety of different computers.
- *Back up your data* This may sound a bit cliché, but frequent and complete backups are one of the most foolproof and reliable means of protecting your vital data. No virus can destroy the backup. Even though the backup may *contain* a virus, it is better to restore an infected backup (and then clean it immediately) than forego the backup entirely. The problem with backups is frequency—how often should it be done? That really depends on how often you use your system. Businesses with active, rapidly changing databases should back up their data at least daily. Casual home users who use only a few utilities infrequently would probably receive little benefit from frequent backups. Most small offices and home offices would be well served to back up every month or so. If new applications or data files are changed dramatically in the meantime, the backup can be updated as needed. The yardstick is simple enough: "if my hard drive were erased now, would I be able to restore it and move on?" When the answer to that question is "no," it's time to back up the system. If the content of your system changes frequently, it may make sense to keep several generations of backups. That way, if Thursday's backup doesn't have the files you need, maybe Monday's *will*.
- *Keep your original disks write-protected* While write protection is not foolproof, it can prevent an infected system from spreading its infection to the disks and thereby proliferating to other systems. This protection can be doubly important for original program distribution disks.
- *Keep an eye out for mysterious or hidden files* While most modern drive utilities have no trouble revealing hidden files, some virulent code may indeed be saved with hidden file attributes. Also check batch files before running them to be sure that there are no destructive commands (such as FORMAT C:).
- *Beware of famous dates* Time bombs often trigger on holidays such as Christmas, New Year, July 4th, or other famous holidays or dates. The day before a special day, set the system clock to the day *after*. For example, on July 3rd, set the system calendar to July 5th. After the holiday has passed, you can easily reset the clock to the correct date.
- *Keep a bootable disk on-hand* Before trouble strikes, invest about five minutes and make yourself a clean bootable floppy disk. The disk should also have a copy of FORMAT, FDISK, DEBUG, PKUNZIP (or your favorite decompression utility), and any other DOS utilities that you need during startup. Be sure to write-protect the floppy disk and keep it in a safe place.

Recognizing an Infection

As any doctor will tell you, the first step toward recovery is *diagnosis*—recognizing the subtle (and not so subtle) signs of viral activity can give you an edge in stopping the activities of a virus and save you a *substantial* amount of time in needless hardware troubleshooting. The following part of this chapter illustrates some of the more important signs of virus activity:

- *You see a warning generated by a virus scanner* Your antivirus package has detected a virus either in memory or in one or more executable files. More comprehensive tools may flag infected e-mail

attachments or prevent Java/ActiveX web items. Once the antivirus package has completed its infection report, go ahead and attempt to disinfect as many files as possible. Many of today's viruses cannot be removed without damaging the executable file, so be prepared to restore the infected files from a backup or original installation disks. After the system is cleaned (and damaged files restored), go ahead and check for viruses again. Repeat this procedure until the entire system is clear.

- *You see some sort of bizarre message (for instance, "legalize marijuana" or "your computer is stoned")* Unfortunately, when a virus reveals itself in this way, it has probably already done its damage to your system. Launch your antivirus software as soon as possible and remove any occurrences of the virus. Be prepared to restore damaged executable files and corrupted data files.
- *You notice that your machine is acting strangely for no apparent reason* This may happen especially on holidays and other important days of the year. Applications may freeze, crash, or produce unusual error messages without warning. You may notice excessive or random disk access where there was none before. The system may behave unusually slowly—files and programs may take a long time to load. Familiar applications may not respond to the keyboard or mouse properly. Leave the application as soon as possible and run your antivirus tools.
- *The computer starts to boot, but freezes before displaying a DOS prompt* Chances are that you've got a command processor infection. Boot the system from a clean, write-protected floppy disk, then try switching to the infected hard drive. If you cannot access the hard drive, it may be defective, or the virus may have affected the drive's partition table. Run an antivirus package to check the system and eliminate any virulent code. When the system is clean, try a drive maintenance package such as DrivePro from MicroHouse to check and rebuild any corrupted boot sector/partition table data.
- *Programs and data files become erased or corrupted without warning* This is a classic sign of a virus at work. It is highly unlikely that the random loss of a single file is due to a hardware defect. DOS drive access works in terms of clusters, and most files require several clusters. If a cluster—or a sector within that cluster—were to fail, the file would still appear in the directory. Run your antivirus package and check for viruses in memory as well as on disk.
- *You see an error message indicating a problem with the file allocation table or the partition table* While this may indeed be the result of a hard drive fault, you should make it a point to boot the system from a write-protected floppy disk and check for viruses. If the system checks clear, go ahead and try a package like Drive Pro by MicroHouse to check and reconstruct the damaged boot areas.
- *Programs access more than one disk drive where they did not before* It is exceptionally rare for a program to try accessing more than one drive unless it is explicitly instructed to do so by you. For example, if you save your new word processing document to drive C:, there will be no reason for the program to access drive A:. Attempts to access an unexpected drive suggest that a virus is trying to slip its operations into normal disk access activities. Leave your application and run a virus checker.
- *The number of bad disk sectors increases steadily* It is not uncommon for viruses to create bad disk sectors and hide within them to escape detection. Since DOS is designed to step over bad sectors, some antivirus programs will not detect viruses using that tactic—leaving you to back up as much of the drive as possible and perform a new low-level format of the drive. Before resorting to that tactic, however, try an antivirus package.
- *The amount of available system RAM suddenly or steadily decreases* DOS provides the MEM function, which allows you to peek at conventional, upper, extended, and expanded memory. If you find that certain programs no longer have enough memory to run, consider the possibility of a mem-

ory-resident virus or replicator or some sort. Try your antivirus package. If you have a memory-resident antivirus product available, try loading that on the system for a while.

- *Memory maps (such as the DOS MEM function) reveal strange TSRs not loaded by CONFIG.SYS or AUTOEXEC.BAT* You can use the MEM function to reveal any drivers or TSRs loaded in the system. If you see a strange or unexpected TSR, you may be faced with a memory-resident virus. Run your antivirus package. If you have a memory-resident antivirus product available, try loading that on the system for a while.
- *File names, extensions, attributes, or date codes are changed unexpectedly* This is another classic sign of viral activity that is usually attributable to older virulent code, which lacked the sophistication to hide its own actions. A reliable antivirus program should be able to deal with any viruses effectively.
- *Unknown files mysteriously appear* This is a tough call for technicians new to a system, but as a computer user, you are generally pretty aware when a new data file is created on your own system (such as a new word processor document or a new spreadsheet). However, when unknown executable files are created, a virus may be at work. Newly created files may be hidden, so use a directory tool that displays hidden files (such as Windows Explorer). Try your antivirus software to locate and eliminate potential viruses.

Other Typical Symptoms

- Your programs take longer to load.
- Your programs' sizes seem to change.
- Your disk runs out of space.
- The CHKDSK report shows less than 655360 bytes available.
- You encounter 32-bit errors under Windows.
- You cannot access the hard drive when booting from the A: drive.
- You notice odd or strangely labeled files.
- You notice clicking noises coming from the keyboard.
- Letters look like they're falling to the bottom of the screen.
- Your computer doesn't retain CMOS settings, but the battery is new.

None of the symptoms here "guarantee" the presence of a virus and may often be caused by other harmless system conditions.

Virus Naming Conventions

When searching for a modern virus name (especially when checking a virus database), you should be aware of the naming conventions used by antivirus tools such as Norton AntiVirus. Virus names consist of a prefix, the name, and often a suffix:

- The *prefix* denotes the platform on which the virus replicates—or the type of virus. DOS viruses usually do not contain a prefix.
- The *name* is the family name of the virus.

- The *suffix* may not always exist. Suffixes distinguish between variants of the same family, and are usually a number denoting the size of the virus or a letter.

Virus names are usually formatted as *prefix.name.suffix*. For example, WM.Cap.A would be the "A" variant of the "Cap" family. The "WM" means the virus is a Word Macro virus. The prefixes in Table 48-3 should help you in searching for viruses.

TABLE 48-3 MODERN VIRUS PREFIXES

A97M	Access Macro viruses that replicate in Access 97.
AM	Access Macro viruses that are native to Access 95.
AOL	Trojans that are specific to America Online environments and usually steal AOL password information.
HLLC	High Level Language Companion virus. These are usually DOS viruses that create an additional file (the companion) to spread.
HLLO	High Level Language Overwriting virus. These are usually DOS viruses that overwrite the host file with viral code.
HLLP	High Level Language Parisitic virus. These are usually DOS viruses that attach themselves to host files.
Java	Viruses that are written using the Java programming language.
PWSTEAL	Trojans that steal passwords.
Trojan/Troj	These files are not viruses, but Trojan Horses. *Trojan Horses* are files that masquerade as helpful programs, but turn out to be malicious code. Trojan Horses do not replicate.
VBS	Viruses that are written using the Visual Basic Script programming language.
W32	32-bit Windows viruses that can infect under all 32-bit Windows platforms.
W95	Windows 95 viruses that infect files under the Windows 95 operating system. Windows 95 viruses often work in Windows 98 also.
W97M	Word 97 Macro viruses. These are native to Word 97 and replicate under Word 97 only.
Win	Windows 3.x viruses that infect files under the Windows 3.x operating systems.
WM	Word Macro viruses that replicate under Word 6.0 and Word 5 (Word 7.0). They may also replicate under Word 97 (Word 8.0), but are not native to Word 97.
WNT	32-bit Windows viruses that can infect under the Windows NT operating systems.
X97M	Excel Macro viruses that are native to Excel 97. These viruses may replicate under Excel 5.0 and Excel 95 as well.
XF	Excel Formula viruses use old Excel 4.0 embedded sheets within newer Excel documents.
XM	Excel Macro viruses that are native to Excel 5.0 and Excel 95. These viruses may replicate in Excel 97 as well.

Dealing With an Infection

Even with the best antivirus tools, regular testing, and consistent backups, systems can still be susceptible to the ravages of computer viruses. When dealing with viruses, you must understand what can and cannot be infected. *Programs* can be infected—*that's all* (though macro viruses can "infect" data files such as documents or spread sheets). Programs are any file that has an extension of .EXE, .COM, .BAT, .SYS, .BIN, .DRV, .OVL, .DLL, .VXD, and of course the two hidden system files that compose the DOS kernel. With the rise of macro viruses, data files such as Microsoft Word and Excel files can also be infected—spreading their havoc when the file's macro is run. Other data files such as images certainly can be corrupted, damaged, or completely destroyed, but they cannot be infected. For example, if you download an Internet image (such as a .JPG file), it cannot contain a virus. It is not impossible to infect programs inside an archive (such as ZIP, ARC, ARJ, LZH, or ZOO files), but it is *extremely* unlikely since a virus does not want you to know it's there—but the programs may have been contaminated before being placed in the archive. When a you suspect the presence of a virus in the system, the following procedures can help you optimize the "damage control":

1. *Boot from a clean, write-protected floppy disk* One of the most fundamental rules of virus defense is that a virus is harmless until it is launched by the boot sector, command processor, or application. If you can *prevent* the virus from loading in the first place, you stand a good chance of running an antivirus tool successfully. Make sure that the boot disk is prepared on a virus-free PC. The disk should also contain a copy of your antivirus package (most are designed to run from a floppy disk). Do not attempt to launch applications from the questionable hard drive until it has been checked and cleaned.
2. *Use your antivirus tools* If the system booted properly from your write-protected floppy disk, the virus(es) in your system should now be neutralized. Start the antivirus tool contained on your floppy disk and run a comprehensive test of all system files. Also make it a point to check the boot sector and command processor. If your current tool does not support boot sector or command processor testing, you should consider using a second tool that does. When viruses are detected (chances are that more than one file will be infected), attempt to remove as many instances as possible. With luck, you can remove viruses without damaging the infected file, but this is often not possible with today's viruses. When a file can not be "cleaned," it should be erased. Be sure to log each erased file and directory path so that you can replace only those files rather than restore entire subdirectories.
3. *Start a quarantine on your computer* Since many viruses propagate by infecting floppy disks, any disks that have been in your computer should be *assumed* to have the virus on them. By assuming the worst case situation, you are possibly saving many others from getting and spreading the virus even further. Gather up as many disks as you can find and check each for viruses. Also, do not share disks between other systems until your system has run for a while and proven itself to be virus-free.
4. *Restore the backups* It is very likely that you had to destroy one or more executable files. Systematically reload any files that were erased during the cleaning process. In most cases, you can restore the damaged files from their original, write-protected installation disks. A tape backup is another popular backup source. Try to avoid reinstalling the entire application unless there is no other alternative.

5 *Recheck the backup* After the deleted files have been destroyed, it is vitally important to restart your antivirus tool and check the suspect disk again. It is not uncommon for recent backups to be contaminated as well. Verify that the drive is still virus-free. If you locate new viruses introduced in the restored files, remove the viruses again and restore the files from original, write-protected floppy disks.

6 *Minimize the collateral damage* Immediately notify anybody who you have given any software, sent e-mail (with attachments), shared bootable disks, or even read their disks on your computer. If you have uploaded any programs to a BBS or the Internet, notify the sysop or webmaster of that system immediately.

LEARNING ABOUT SPECIFIC VIRUSES

There are tens of thousands of computer viruses in the field today—each with its own aliases, modes of infection, and techniques for removal. It would be impractical to index all of that information here. Fortunately, most major antivirus makers (such as McAfee and Symantec) provide extensive virus "encyclopedias" over their Internet web sites. If you can get online, you can easily find detailed information on just about any virus or strain and often learn the specific procedures for removing the particular virus.

Understanding Antivirus Tools

As the awareness of computer viruses grew through the last decade, so did the proliferation of antivirus tools designed to combat the threat. However, you should understand that every antivirus tool is created as a *response* to viruses that have *already* penetrated the PC environment. As a result, antivirus products are forever playing "catch-up" with ever-more sophisticated virus programmers. No antivirus product is 100 percent effective in *all* forms of detection. The one rule to remember with *all* antivirus tools is that they become outdated very quickly. As a technician, you must make it a point to keep your antivirus tools current. In the perpetual virus "arms race," you should seriously consider updating any product over six months old. This part of the chapter examines the major antivirus tactics and explains the limitations of each approach.

VACCINES

These were the earliest form of virus protection and acted by appending small programs and checksums to various executable files. Whenever the "vaccinated" program is run, the antivirus vaccine calculates the program's checksum and compares it to the appended checksum. If the two checksums match, control is returned to the executable file and it runs normally. When the comparison fails because of file damage or the presence of a virus, a warning is generated and corrective action can be taken. There are a number of serious drawbacks to the vaccine technique that you should be familiar with:

- The vaccine (or *antigen* as it was called) is little more than a virus itself. Although it does not reproduce without permission or harm files, many users felt uncomfortable "inoculating" their files intentionally.
- When there are a large number of executable files, the increased disk space needed for each appended vaccine can become significant.
- Device drivers, overlay files, packed EXE files, and executable data files cannot be vaccinated.
- False alarms are common—especially for self-modifying programs like Borland's SideKick—which force users to remove vaccine protection.

- In some cases, the modifications to an executable file in order to vaccinate it can cause unpredictable program operation—some programs simply do not work with vaccine-based viral defense.
- The virus-type behavior of vaccines often cause false alarms with other non-vaccine antivirus programs.
- Since vaccine techniques are the same for every files, it is a simple matter for a virus to bypass the vaccine's loading checksum test—so vaccines provided limited viral protection.

FILE COMPARISONS

This is a plain and simple technique that utilizes byte-by-byte comparisons between known-good files and potentially infected files. Any variation between the two signals the possibility of a virus. File comparison techniques were initially embraced because they were easy to develop and quick to document, so they were an inexpensive option for antivirus developers. However, file comparison presents some serious disadvantages in the marketplace:

- The most critical problem is the need for known-good files to be added to the disk (in addition to the normal operating files). Even for large drives, this is a hideous waste of valuable disk space.
- File comparison antivirus tools often lack the typical resources that are considered to be standard equipment for virus management (such as activity logs, data encryption, comprehensive warnings indicating which virus is at work, system lockouts, and wildcard file searches).
- It is a simple matter for viruses to search a disk looking for multiple copies of a file and infect *both* copies—rendering the file comparison technique useless.

ANTIDOTES

Software antidotes (sometimes called *disinfectors* or *eradicators*) are a close cousin to vaccines—they operate by "surgically removing" the virus. But antidotes are designed specifically to deal with a limited set of viral strains within a small group of program types. Often, an antidote is designed to check and remove a particular virus. For example, the media scare surrounding the Michaelangelo virus some years back resulted in a number of related "antidote" products developed specifically to check for and eradicate the virus. Such limited operation presents several serious limitations:

- The limited nature of antidotes makes them unsuited for general, systemwide use. Viruses not specifically addressed by the antidote remain totally untouched.
- Since viruses are constantly changing, antidotes must continuously be updated and expanded—otherwise, the antidotes quickly become useless. The constant expense of regular updates is often too much for the average computer user.
- Antidotes often destroy program files while trying to remove virulent code and are reputed to suffer frequent false alarms that cause the antidote to alter good files in an attempt to remove a virus that is not there. Effectively, this destroys good files as well.
- Each executable file has its own particular characteristics and internal structure. As a result, it is virtually impossible for any one antidote to remove a virus from every possible file type.
- Generally, it is safer and more reliable to recover an infected file by overwriting it with an uninfected copy rather than trust an antidote to surgically remove the virus.

SIGNATURE SCANNERS

Currently, the virus scanner is the most widely accepted type of antivirus tool. Scanning basically checks each executable file against a fixed set of *virus signatures*, tell-tale fragments of code that indicate the presence of particular viruses. When the virulent code is identified, it can be removed fairly accurately, but many executable files will still be destroyed. The technique is fast and flexible, viruses can be identified very accurately, and there are few instances of the false alarms or incompatibilities that older techniques suffer from. However, there are still limitations to virus scanning:

- Scanners rely on a *fixed* set of signatures—if a signature is not in the database, it is not checked. Signature databases are easy to update, but the updates can often be costly (though now easily downloaded from the Internet). Since viruses are constantly changing, signature databases become outdated quickly.
- Virus scanners cannot detect signatures that change or mutate as the infection propagates through the system. As a result, scanners are largely ineffective against stealth or polymorphic viruses (though this is changing as virus scanners are integrating newer encryption detection techniques).

MEMORY-RESIDENT UTILITIES

One breed of antivirus tool can be loaded into memory where it will remain resident (TSR) and provide "last-minute" protection against viral infiltration of disk commands and other viral activity. Unfortunately, this class of antivirus tool suffers from a set of very serious problems:

- As a TSR, the program must remain in memory, consuming valuable amounts of memory (often significant) that are needed by other applications. It is not uncommon to eventually disable TSRs to free extra memory for large applications.
- False alarms are commonplace with antivirus TSRs, which mistake disk caching or normal system activity with virus activity. Even communication functions such as e-mail downloads are often interrupted as virus attacks.
- Many systems respond poorly to TSRs. When you consider that TSR technology is intended to coerce DOS into perform multitasking—a feature it was not intended to do—it is no wonder that TSR development is nonstandardized. As a consequence, TSRs are often quite troublesome. When used with combinations of other device drivers and TSRs, antiviral TSRs can present a serious problem.
- Viruses can circumvent antivirus TSRs by accessing PC hardware directly (such as direct access of disk controllers).

DISK MAPPERS

The disk mapping technique is similar to the file comparison process. A mapper maintains a single data file that contains a coded "snapshot" of the protected disk. Each time a mapper is run, it notifies you about any variations between the protected disk files and the "key map." Ideally, these variations will alert you to the possibility of a virus. Many later disk mapping schemes allow users to specify exactly which files (or file types) must be monitored. However, this benefit is not enough to overcome some inherent problems:

- Creating a "key map" of the disk can require a substantial amount of space. The space demand increases along with the number of files that must be "mapped."
- For most professional users, the state of a PC is changing constantly as files are created, modified, and deleted. This constant change demands regular maintenance of the "key map." Such maintenance is often cumbersome and time consuming, since disk mappers are typically complex systems to use.

- Disk mappers are typically tied into the boot process to ensure regular "key map" checks and updates. This process results in longer (sometimes *much* longer) boot times.
- Disk mappers are not immune to infiltration and damage by viruses. Some viruses seek out and destroy "key map" files.

MODERN ANTIVIRUS COMPONENTS

Today, antivirus software is much more than just a simple command-line–driven utility. Modern antivirus software is actually a combination of powerful interrelated tools that each serve a specific purpose on your system. This part of the chapter examines the components of a current antivirus software package, such as McAfee's VirusScan.

This discussion is for example purposes only. Your own antivirus software may use more or fewer components, but the basic suite of features will probably be quite similar.

- ***Basic components*** Also called "common" components. These are the data files and other support files that are shared by the antivirus package. They includes DAT ("signature") files, default configuration files, validation files, online Help, and so on.
- ***Command line scanner*** This tool lets you scan for viruses when Windows 95/98 won't start, so you can add it to your bootable or "emergency backup" disk. For example, McAfee's VirusScan offers two such tools: SCAN.EXE and BOOTSCAN.EXE. BOOTSCAN.EXE normally runs as soon as you start your system—it checks for viruses that hide in the boot sectors on your hard disk or that load themselves into memory during the boot process. SCAN.EXE is an independent program that can scan your system from the DOS prompt. Its low resource requirements allow you to fit both SCAN.EXE and boot files onto a single floppy disk. You control the activities of a command line scanner through the use of command line switches added to the command line when the program is launched.
- ***Protected-mode (GUI) scanner*** This is the heart of your antivirus package, It provides your scanning capability under Windows 95/98/NT, for instance, and is the main tool that you can use to scan files, disks, and network targets "on demand." You can also control how the scanner will respond to infection reports and report its activities back to you.
- ***Scanner interface*** This is typically a simple interface that allows you to access and control each of the other components, that generates statistics, and that displays reports or other information. You can also use the scanner interface as the conduit to update your antivirus data files.
- ***Memory-resident scanner*** This tool gives you continuous antivirus protection from viruses carried on floppy disks, brought in from your network, or loaded into memory. The memory-resident scanner (for example, McAfee's VShield) starts when you boot your computer and stays in memory until you shut down the system. You usually tell the memory-resident scanner what parts of your system to scan, when to scan them, which to leave alone, and how to respond to any infected files it finds. The latest versions of memory-resident scanners also include Java and ActiveX protection and can even prevent access to potentially dangerous Internet sites.
- ***Mail scanner*** This type of tool (such as McAfee's cc:Mail Scan) will scan Lotus cc:Mail mailboxes that do not use Microsoft's Messaging Application Programming Interface (MAPI) standard. This tool is for a workgroup or network that uses cc:Mail v7.x or earlier.
- ***MAPI scanner*** Here's a tool that allows you to scan an inbox or other mailbox for e-mail client applications that adhere to Microsoft's Messaging Applications Programming Interface (MAPI). Use it to supplement the continuous background scanning of your memory-resident scanner.

- ***Background scan scheduler*** This tool if present in your package can scan your computer as your screen saver runs during idle periods.
- ***Scan scheduler*** Similar to Windows 98's Task Scheduler, a scan scheduler lets you assign scanning tasks for your virus scanning software.

TESTING A VIRUS CHECKER

Once you install your antivirus software, it should be ready to scan your system for infected files. However, you can test whether it is installed correctly and verify that it can properly scan for viruses by using this test developed by EICAR (a coalition of antivirus vendors headquartered in Europe):

1. Open a standard Windows text editor (such as Notepad), then type the following line:

   ```
   X5O!P%@AP[4\PZX54(P^)7CC)7}$EICAR-STANDARD-ANTIVIRUS-TEST-FILE!$H+H*
   ```

The line shown above should appear as one line in your text editor.

2. Save the file with the name EICAR.COM. The file size should be 69 or 70 bytes.
3. Start your antivirus software and allow it to scan the directory that contains EICAR.COM.
4. When it examines this file, it will report finding the EICAR-STANDARD-AV-TEST-FILE virus.
5. If the "virus" is found, then you know the antivirus tool is installed and working.

This EICAR.COM file is *not* a virus—it cannot spread or infect other files or otherwise harm your system. Simply delete the file when you have finished testing your installation.

UNDERSTANDING "FALSE DETECTIONS"

A "false detection" occurs when your antivirus software sends a virus alert message (or makes a log file entry) that identifies a virus where none actually exists. You're more likely to see false detections if you have antivirus software from more than one vendor installed on your computer, because some antivirus software stores the code signatures it uses for detection unprotected in memory. The *safest* course to take when you see an alert message or log entry is to treat it as a genuine virus threat and to take the appropriate steps to remove the virus from your system. But if you believe that the antivirus software has generated a false detection (that is, it flags a file as "infected" when you have used it safely for years), check for one or more of the following situations before you contact the antivirus software maker:

- You have more than one antivirus program running. If so, one of the scanners might detect unprotected code signatures that another program uses and report *them* as viruses. To avoid this problem, configure your computer to run only one antivirus program, then shut the computer down and turn off the power. Wait a few seconds before you start the computer again so that the system can completely clear the other program's code signature strings from memory.
- You have a BIOS chip with antivirus features. Many current BIOS versions provide antivirus features (intended to prevent boot sector infection) that can trigger false detections when antivirus software runs. You can try disabling the antivirus protection in your BIOS through the CMOS Setup.
- You have an older Hewlett-Packard or Zenith PC. Some older models from these manufacturers modify the boot sectors on their hard disks each time they start up. Your antivirus software might detect these modifications as viruses when, in fact, they are not. To solve the problem, use the command line

version of your antivirus software to add validation information to the startup files themselves. This method does not save information about the boot sector or the master boot record.

- You have copy-protected software. Depending on the type of copy protection used, your antivirus software might detect a virus in the boot sector or the master boot record on some floppy disks or other media.

If none of these conditions are true, you should contact the antivirus software maker and inform them of the false detection so that they can investigate the matter and make suitable corrections in subsequent patches and data updates.

Troubleshooting Antivirus Tools

The key to dealing with computer viruses is the proper use of antivirus tools. A quick walk through almost any software store will show you just how many antivirus products are available. Being able to use those products properly and successfully is not always a simple task. This part of the chapter offers some guidelines to help you handle problems with the tools themselves.

Although there are no antivirus tools on the accompanying CD, you can easily download current, fully functional demo or shareware antivirus tools from the resources listed at the end of this chapter.

TIPS FOR PREVENTING MACRO VIRUSES

Macro viruses can be detected by most of the antivirus tools now available (and you should regularly scan documents for macro viruses), but you may be able to reduce the risk of macro virus effects with the following tips:

- Mark the NORMAL.DOT template file as "read-only." This generally protects the NORMAL.DOT file from infection.
- Use Word 7.0a or Word 97 from Microsoft. These versions present an Alert box when the file you are going to open contains macros or customization information. You also have the opportunity to disable unknown macros before they operate.

MANUALLY REMOVING A MACRO VIRUS

Chances are that you've received virus alerts for macro viruses in one or more Microsoft Word documents. If you've tried to repair the documents with your antivirus software, but the software has been unsuccessful, you may be able to remove the macro virus manually. The steps below will remove the virus only from existing documents and should be considered an emergency repair. If the virus that's creating the infected macros is still on your system or network, the infection will reoccur the next time the infected document is opened. Please follow these steps to manually remove the macro virus from a document and recover its text:

You'll be making changes to Microsoft Word documents. Before proceeding, make a backup of the suspect documents to clearly labeled media, such as a floppy disk.

1. Click Start, highlight Find, and click Files or Folders. The Find dialog box appears.
2. Type **NORMAL.DOT** and click Find Now.

3. Once the file is found, right-click the file name and click Rename on the shortcut menu.
4. Rename the file to **NORMAL.OLD** and press ENTER.
5. Close the Find dialog box.
6. Start Word—this will recreate the NORMAL.DOT template.
7. Choose the File menu and click Open.
8. Navigate to the folder containing the infected file and select it.
9. Press and hold the SHIFT key and click Open. Continue to hold the SHIFT key until the affected file is open in Word (holding the SHIFT key while opening a file keeps any macros from running).
10. Choose the Tools menu, point to Macro, and then click Macros.
11. In the "Macros In" list box, select "All active templates and documents."
12. Select the suspect macro and click Delete. Click Yes to confirm.
13. Repeat the previous step for all suspect macros.
14. Click Close.
15. Choose the Edit menu and click Select All.
16. Press SHIFT+LEFT ARROW to deselect the last paragraph mark in the document.
17. Choose the Edit menu and click Copy.
18. Choose the File menu and click New. Select the desired template and click OK.
19. Choose the Edit menu and click Paste.
20. Repeat steps 10 to 14 to verify that the viral macros have not replicated.
21. Save the document.
22. Repeat these steps for any document you think may contain a macro virus.

Symptoms

SYMPTOM 48-1 **You can not run more than one antivirus product at a time** This is not an uncommon problem and occurs most frequently when memory-resident virus protectors conflict with file-based antivirus tools. When you run more than one antivirus program, there is always the risk of strange results and false alarms. For example, some antivirus programs store their "virus signature strings" unprotected in memory. Running incompatible or conflicting antivirus tools may detect other signature strings or memory-resident activity as a virus. Run only one antivirus program at a time.

SYMPTOM 48-2 **Your antivirus tool does not function or causes other drivers to malfunction** Some "terminate-and-stay-resident" (TSR) software may conflict with some antivirus programs, especially memory-resident antivirus programs. When problems occur, try booting the system from a clean boot disk so that there are no drivers or TSRs in the system other than the antivirus tool.

SYMPTOM 48-3 **You notice that your antivirus tool is slowing disk access dramatically, or it locks up under Windows** Normally, many antivirus tools (especially memory-resident tools) will slow disk access a bit. When there is a tremendous reduction in disk performance, or the tool freezes during operation, it may be that the disk cache being used conflicts with the antivirus

product. Try increasing the number of buffers in the CONFIG.SYS file. If problems continue, try disabling the disk caching software while running the antivirus product.

SYMPTOM 48-4 **The antivirus tool is reporting false alarms** It is not uncommon for antivirus products to report false alarms. This happens most often because of conflicts with other memory-resident software running in the system. Try running the software from a clean boot disk. The nature of antiviral detection techniques also plays a role in reporting false errors. For example, file comparison is a typical technique, but files can be changed for many reasons other than a virus, so false alarms are a strong possibility. Other techniques also have flaws that may result in false alarms. You should try updating the virus signature database from your antivirus software provider.

SYMPTOM 48-5 **You are unable to remove the memory-resident antivirus tool** There is probably another TSR running in the system that is conflicting with the antivirus tool. You may have to reboot the system in order to clear the antivirus tool. In the future, try loading the antivirus tool last—after all other drivers and TSRs are loaded.

SYMPTOM 48-6 **The virus scanner is scanning files very slowly** This is usually an issue with certain older virus scanning software. Ideally, you should be able to correct this problem by upgrading to the latest patch or version of the virus scanner. If you cannot patch or update the program, try scanning only the "Program files" and not "All files" or "Compressed files."

SYMPTOM 48-7 **The virus scanner seems to conflict with the boot sector when it scans** If the virus scanner is conflicting with your boot sector (either during or after installation) try choosing the "Custom" setup feature and disable the *initial system scan* during installation. Then edit the scanner's configuration to skip the boot scan. As an example for McAfee's VirusScan product, edit your DEFAULT.VSC file and, under the [Scan Options] section, change bSkipBootScan=0 to bSkipBootScan=1. Doing this will make the scanner skip the boot sector when you run VirusScan (the boot sector will not be scanned for viruses).

SYMPTOM 48-8 **You receive a "Cannot Load Device Drivers Error" from the virus scanner** This error typically occurs on platforms that have been upgraded from Windows 3.1 to Windows 95/98, but have not completely uninstalled the 3.1 version of virus scanner (or a previous installation of a Windows 95/98 virus scanner was not completely removed from the system). You'll need to remove all traces of the virus scanner manually from SYSTEM.INI and WIN.INI. Let's use McAfee's VirusScan as an example. Open the SYSTEM.INI file and remove:

```
device=MCSCAN32.386
device=MCUTIL.386
device=mCKRNL.386
device=MCFSHOOK.386
device=vshield.386
```

Open the WIN.INI file and remove:

```
load = C:\MCAFEE\VIRUSCAN\VSHWIN.EXE
```

And remove the section:

```
[VIRUSCAN]
WSCAN=C:\McAfee\VIRUSCAN\WSCAN.EXE
```

Of course, you should be sure to remove the correct entries for your particular virus scanner.

SYMPTOM 48-9 You receive an "Insufficient memory" message when the virus scanner is loading under Windows 95 This error is usually caused when Windows 95/98 uses a DOS version of a virus scanner to scan the root directory of C: at startup, and there is not enough conventional memory to run the DOS virus scanner. Try updating the virus scanner program, or patching it to a later version if possible, or disabling virus scanning on Windows 95/98 startup.

SYMPTOM 48-10 You receive a "Cannot create events" error when the virus scanner is loading This is usually due to an improperly located KERNEL32.DLL file. Search your computer for the file KERNEL32.DLL on the root of your hard drive (C:). Moving this file to C:\Windows\System, where it belongs, should resolve this issue. Some new systems are shipped with the KERNEL32.DLL file improperly located in the root directory.

Further Study

Command Software Systems: **http://www.commandcom.com/**

EICAR: **http://www.eicar.org**

IBM: **http://www.av.ibm.com/**

McAfee: **http://www.mcafee.com or http://www.networkassociate.com/**

NCSA: **http://www.ncsa.com/**

S&S Software International: **http://www.drsolomon.com/**

Symantec: **http://www.symantec.com/avcenter**

VSUM: **http://www.vsum.com**