Viral
Contagia in Cyberspace

More combat casualties come from disease and nonbattle injuries than from enemy fire. In recent years, infections within a single organization have threatened the health and function of the entire Army. What can cause such a terrifying epidemic? Viruses—computer viruses. Deal, Schueneman and Gage discuss causes, cures and preventive medicine that our powerful, far-flung, 21st-century force must understand to avoid frustration or incapacitation by rogue codes.

Colonel John C. Deal, US Army;
Robin Schueneman; and Major Gerrie A. Gage, US Army

THE RECENT denial-of-service attacks against Internet service providers and content providers suggest that computer networks are vulnerable to widespread attack from various adversaries. The global nature of such activities and the disparate attacks complicate providers' defensive tasks.

Dispersed tool kits available to hackers make it all but certain that sniffing out, tracking down and eliminating these threats will frustrate the best network minds. As webmasters, systems administrators and network security managers rethink the problem, they will focus much of their effort on mitigating all forms of virus attacks.

The similarity between computer network systems and biological systems is uncanny. This comparison is common both within information technology publications and among computer network system users. Comparing computer networks to living systems suggests their numerous vulnerabilities. One of the greatest threats to an organization's computer networks is viral infections or contagion. Containing these contagion and eradicating them before a network is degraded requires understanding and real-time vigilance from users, network administrators and software developers.

The Pathology of Computer Viruses

A computer virus is a program, software code, designed to replicate and spread, generally without the victim's knowledge. The mere mention of these two words sends computer novices and experts scrambling to download the latest updates of antivirus software. Without exception, every large company and organization experiences viral infections—many experience them monthly. According to TruSecure's *ISCA Labs 6th Annual Computer Virus Prevalence Survey 2000*, in one two-month period, corporations experienced virus attacks at the rate of 91 infected personal computers (PCs) per 1000, which represents the fifth consecutive year of increase.[1] The number of virus attacks appears to be unusually high if viewed independently. However, when Department of Defense (DOD's) antiviral software supplier defines and categorizes 21,389 known viruses and DOD's other antiviral software supplier categorizes more than 40,000 viruses, the number of virus attacks appears in a new light. These viruses, usually benign or annoying, can slow performance, absorb resources, change screen displays and disrupt or deny service, affecting organizations' profit or mission.

Computer viruses come from a variety of sources and spread by attaching themselves to other programs, such as word processors or spreadsheet applications, or to the boot sector of a disk. When the infected file is activated, or executed, or when the computer is started from an infected disk, the virus is also executed. Viruses can also lurk in computer memory, waiting to infect the next activated program or the next accessed disk.

The TruSecure study found that e-mail attachments were responsible for 87 percent of infections, up from just nine percent in 1996. Infected diskettes, while accounting for 57 percent of infections in 1996 declined to just 8 percent in 2000. About one percent of computer users said they had acquired their virus while downloading software from an electronic bulletin board service or web-site. Other sources of infected diskettes (six percent) included demo disks, diagnostic disks that service technicians use and shrink-wrapped software disks.[2]

In one two-month period, corporations experienced virus attacks at the rate of 91 infected PCs per 1000. . . . DOD's antiviral software supplier defines and categorizes 21,389 known viruses and DOD's other antiviral software supplier categorizes more than 40,000 viruses.

Although no new statistics are available, networking, enterprise computing and interorganizational communications are growing. As telecommuting and networking increase, so do infections—in number, complexity and variety. In 1986 there were just four known PC viruses. Today more than three viruses are created daily for an average of 110 new viruses in a typical month. There are several variations of viruses, but there are only three ways a virus can access a system.[3]

File viruses. Most of the thousands of existing viruses are file viruses, including the Friday the 13th virus. They generally infect files by attaching themselves to executable files, such as the .EXE and .COM files that execute applications and programs. The virus can insert its own code in any part of the file, then changes the host's code and misdirects the program to execute the virus code first rather than the legitimate program.

Boot-sector/partition-table viruses. About 200 different boot-sector viruses make up 75 percent of all virus infections. Boot-sector viruses include the most common virus of all time, Stoned, and the most notorious, Michelangelo. These viruses are so prevalent because they are difficult to detect. They do not change a file's size or slow PC performance and, thus, are invisible until their trigger event occurs. Events such as reformatting a hard disk or scanning a disk serve as triggers. The boot-sector virus infects floppy and hard disks by inserting itself into the boot sector, which contains code that is executed during the system boot-up process. Booting from an infected floppy allows the virus to jump to the computer's hard disk. The virus executes first and gains control of the system boot program code even before the operating system (OS) is loaded.

Because the virus executes before the OS is loaded, it is not OS-specific and can infect any PC operating system platform. The virus goes directly to the random access memory (RAM) and infects every disk that is accessed until the computer is rebooted and the virus is removed from memory. Partition-table viruses attack the hard disk partition table by moving it to a different sector and replacing the original partition table with its own infectious code. These viruses spread from the partition table to the boot sector of floppy disks as floppy disks are accessed.

Multipartite viruses. These viruses combine the ugliest features of both file and boot-sector/partition-table viruses. They can infect any of these host software components. While traditional boot sector viruses spread only from infected floppy boot disks, multipartite viruses can spread with the ease of a file virus but still insert an infection into a boot sector or partition table. This makes them particularly difficult to eradicate. Tequila is an example of a multipartite virus.

Although there are only three ways to infect a system, there are hundreds of variations of viruses.[4] Like its classical namesake, the Trojan Horse virus typically masquerades as a legitimate software program. The Trojan Horse generally does not replicate; it waits until its trigger event and then displays a message or destroys files or disks. Alongside the Trojan Horse is the Trojan Mule. The Trojan Mule fools authorized users into giving their LOGIN information—passwords and user-IDs. Once they type in their user-ID/password LOGIN information, the virus sends that information to the file implementers and prints out a LOGIN error message. As the authorized user retypes the information, the virus exits, and the real LOGIN program regains control without the user suspecting the LOGIN information has been revealed. The difference between Trojan Horse and Trojan Mule is that the mule does not even try to perform useful functions, such as games or applications. The mule disappears from the system it has infected, but the horse remains until it is cleaned out.

File overwriters. These viruses infect files by linking themselves to a program, keeping the original code intact and adding themselves to as many files as possible. Innocuous versions of file overwriters may not be intended to do anything more than replicate, but even then they take up space and slow performance. Since file overwriters, like most other viruses, are often flawed, they can dam-

age or destroy files inadvertently. The worst file overwriters remain hidden until their trigger events, then deliberately destroy files and disks.

Polymorphic viruses. More and more of today's viruses are polymorphic. The recently released Mutation Engine—which makes it easy for virus creators to transform simple viruses into polymorphic ones—ensures that polymorphic viruses will proliferate over the next few years. Like the acquired immunodeficiency syndrome (AIDS) virus that mutates frequently to escape detection by the body's defenses, the polymorphic computer virus mutates to escape detection by antivirus software that compares it to an inventory of known viruses. Code within the virus includes an encryption routine to help the virus hide, plus a decryption routine to restore the virus to its original state when it executes. Polymorphic viruses can infect any type of host software. Although polymorphic file viruses are most common, polymorphic boot sector viruses have already been discovered.

As telecommuting and networking increase, so do infections—in number, complexity and variety. In 1986 there were just four known PC viruses. Today more than three viruses are created daily for an average of 110 new viruses in a typical month.

Stealth viruses. Stealth viruses have special engineering that enables them to elude detection by traditional antivirus tools. The stealth virus adds itself to a file or boot sector, but when you examine the host software, it appears normal and unchanged. The stealth virus performs this trick by lurking in memory when it is executed. There it monitors and intercepts the OS's calls. When the OS tries to open an infected file, the stealth virus races ahead, disinfects the file and allows the OS to open it—all appears normal. When the OS closes the file, the virus reverses these actions, reinfecting the file. Boot sector stealth viruses insert themselves in the system's boot sector and relocate the legitimate boot-sector code to another part of the disk. When the system is booted, it retrieves the legitimate code and passes it along to accomplish the boot. The boot-sector appears normal, but is not in its normal location.

Macro viruses. Macros are miniprograms that take much of the legwork out of repetitive or template-oriented documents. For example, to minimize the work involved in typing out the current date, a user might simply type the letter "D" to insert the day, month and year all at once. Macro viruses are carried in the types of data files that business computer users most often exchange— word-processed documents and spreadsheets. These data files are often exchanged by e-mail, so they sometimes bypass the checks that virus-aware organizations already have in place. Experts estimate that 40 percent of virus attacks are made this way.

Macro viruses are created with the aid of the macro routines contained in all word-processing and spreadsheet application software. They attach themselves to any document files that include the macrocode so they can be executed through the application software. The purpose of macro languages is to insert useful functions into documents that are then executed as the documents are opened. This is what makes macro viruses easy to write. One reason they have become so prevalent is the success of Microsoft Office, which has 80 percent of the global market for integrated packages—a tempting target for macro virus writers.

Memory-resident viruses. This characteristic is the most common among viruses. When viruses load into memory via a host application, they remain in memory until the computer is turned off. This stage of existence allows easy replication into subsequently launched boot sectors or applications.

Nonmemory-resident viruses. These viruses can infect a system only when the host application is running. Closing the host application also closes down the virus. Therefore, opening applications after closing a host application cannot infect the system with that specific virus.

Companion viruses. Understanding this characteristic requires understanding the sequential order in which system files work. When launching an executable file, users either manually issue commands or have the interface execute them. Most applications have a file-type (FT) extension of .EXE. When invoking these commands, the user or the computer enters the name of the application without the extension. The computer executes other system files with the same name before executing the .EXE applications FT. A companion virus creates a name that matches the .EXE file name but with a different extension such as .COM. The .EXE still executes; however, the .COM (infected file) launches first and infects the system. Most antiviral software packages can identify this characteristic.

Bomb. This type of Trojan Horse releases a virus, a worm or some other system attack. It is either an independent program or a piece of code that a system developer or programmer has planted. A bomb triggers some kind of unauthorized action

when a particular date, time or condition occurs. There are two types of bombs: time and logic. A bomb that is set to go off on a particular date or after some period of time has elapsed is a time bomb. Friday the 13th was a time bomb. A bomb set to go off when a particular event occurs is a logic bomb. Software developers have been known to explode logic bombs at key moments—for example, if a customer fails to pay a bill or tries to make an illegal copy.

Spoof programs trick the unsuspecting users into giving away privileges. Often the spoof is perpetrated by a Trojan Horse mechanism in which an authorized user is tricked into inadvertently running an unauthorized program. The program then takes on the privileges of the user and runs amok.

Bacteria programs do nothing but copy themselves, but by doing so, they eventually use all system resources such as memory and disk space.

Rabbits are rapidly reproducing programs.

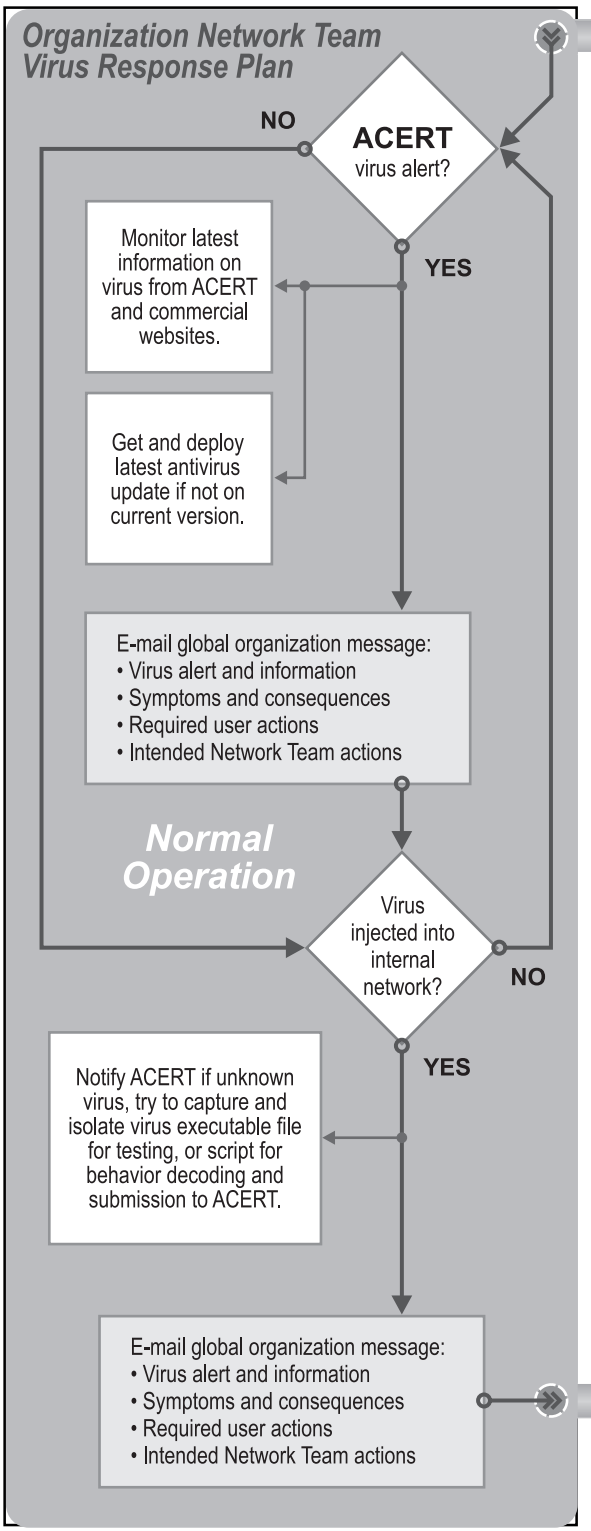
Crabs attack the display of data on computer terminal screens.

Salami slices (rather than hacks) away tiny pieces of data. For example, a salami alters one or two numbers or a decimal point in a file or shaves a penny off a customer's bank interest calculations and deposits the penny in the intruder's account.

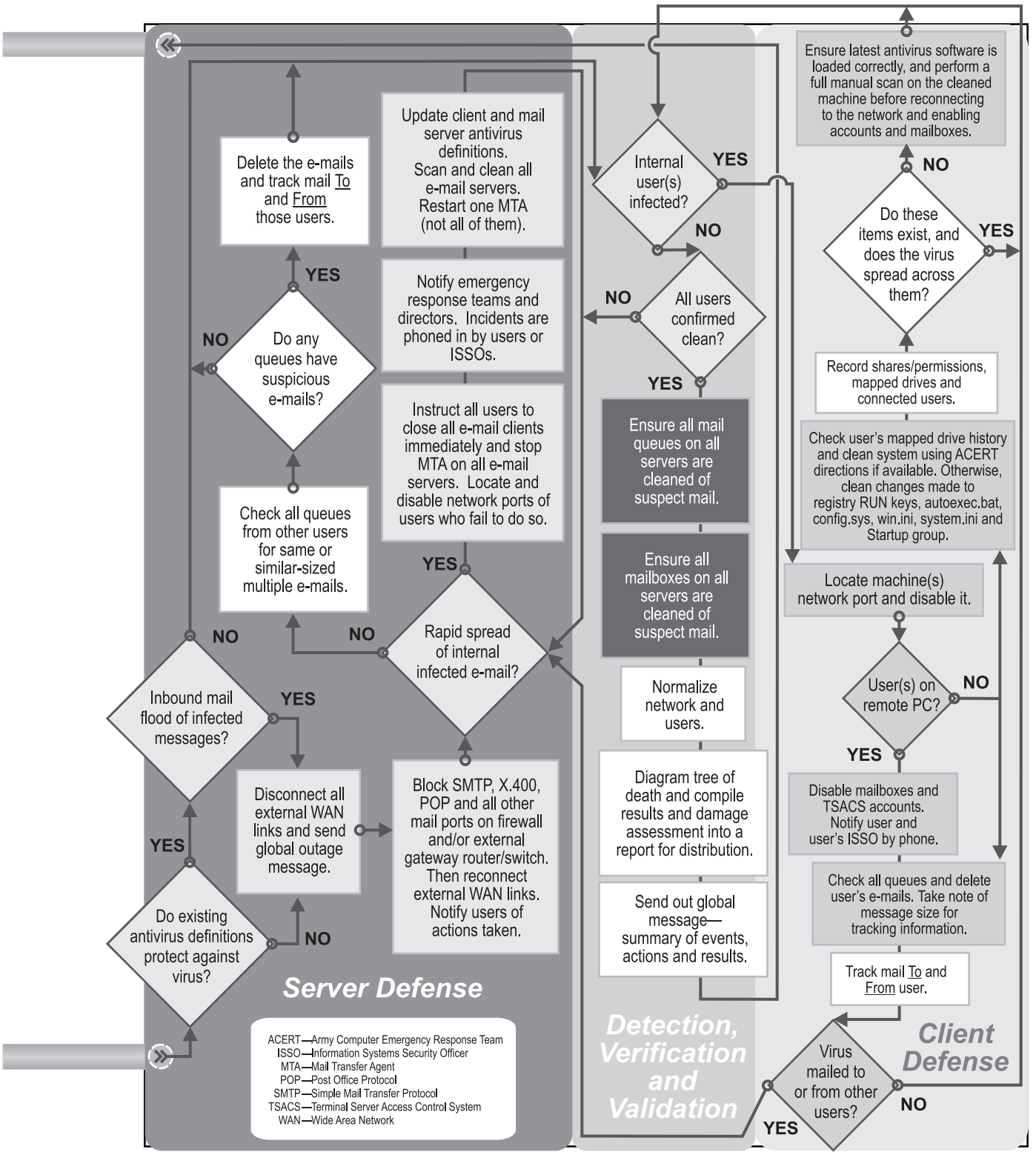
Viruses affect computers and networks differently. Viruses are intended to remain undetected and spread throughout the organization until they degrade performance or destroy data. Most viruses give no symptoms of their infections, thus driving the use of antivirus tools. Antivirus tools allow users to identify these quiet killers. However, some viruses are flawed and provide some tip-offs of infection. Watch for these indications:

- Changes in the length of programs.
- Changes in the file date or time stamp.
- Longer program load times.
- Slower system operation.
- Reduced memory or disk space.
- Bad sectors on your floppy.
- Unusual error messages.
- Unusual screen activity.
- Failed program execution.
- Failed system bootups when booting or accidentally booting from the A: drive.
- Unexpected writes to a drive.[5]

This list of virus variations and symptoms is not all-inclusive.[6] These are only the most common variations of computer viruses and their symptoms. Computer viruses have cost companies worldwide nearly $2 billion since 1990, with costs accelerating by another $1.9 billion in 1994 alone. This cost is directly related to virus cleanup, not profit loss, which is impossible to calculate. Organizations are combating the virus problem with antivirus software. The software cost is expected to grow from $700 million in 1997 to $2.6 billion by 2001.[7]



What can organizations do to prevent computer viral infections, and what is the best response in



the event of an infection? Analysis of a real event provides the best answers. The US federal government, DOD and industry responded impressively to infection according to System Administration Networking and Security.

Containing Contagion: A Case Study

The mythical god of love, Cupid, shot a poison-tipped arrow felt around the computer world. His aim was true, attacking millions of commercial, military, educational and home PCs with the I Love

You virus. On 3 May 2000, the first "socially engineered" worm was released on the computer world. By 4 May 2000, more than 350,000 computers' files were infected in the United States, 30 percent of Britain's business e-mail systems were brought down, Sweden experienced an 80-percent business e-mail systems effect, and there were reports from 18 other countries.[8]

A worm "reproduces by copying itself from one computer to another, usually over a network. Like a virus, a worm compounds the damage it does by spreading rapidly from one site to another; however, instead of spreading from file to file like a virus, worms spread from computer to computer, infecting an entire system. Also, a worm does not destroy data like a virus; a worm typically does its damage by harnessing resources in a network by tying up these resources and eventually shutting down the network."[9] This was a denial-of-service attack.

I Love You was not a particularly complicated or sophisticated worm, and there were no original components in its code, but it combined several techniques, thus becoming extremely destructive. . . . [causing] an estimated $2.6 billion worth of damage.

More precisely I Love You was a visual basic script (VBS) worm with overwriting VBS virus-like qualities that spread through e-mail as a chain letter and attempted to use companion techniques by adding a secondary file next to the existing file.[10] I Love You was not a particularly complicated or sophisticated worm, and there were no original components in its code, but it combined several techniques, thus becoming extremely destructive. The social engineering of the I Love You worm made it subtle and menacing. First, the worm disguised itself as a love letter, which recipients were pleased and usually eager to receive. The letter was probably from someone they knew, and they opened the e-mail attachment without thinking of a potential infection.

It also added an extra extension onto files, but the second extension was usually hidden in Windows so it appeared to be a simple text file.[11] When the attachment was executed, it first copied itself to the Windows system directory and to the Windows directory. Then it added itself to the registry so it would be executed when the system was restarted. Then the worm replaced the Internet Explorer home page with a link that pointed to an executable program, "WIN-BUGSFIX.exe." If the file was downloaded, the worm added this to the registry as well, causing the program to be executed when the system was restarted.[12]

The worm then creates a Hyper Text Markup Language (HTML) file, "LOVE-LETTER-FOR-YOU.HTM," in the Windows system directory. This file contained the worm, and it would be sent using the Mardam-Bey's Internet Relay Chat (mIRC) whenever the user joined an IRC channel. The worm then used Outlook to mass mail itself to everyone in each address book, and after the mail had been sent, it added a marker to the registry and did not mass mail itself again. The virus then searched for certain file types in all folders on all local and remote drives and overwrote them with its extension codes—vbs, vbe, js, jse, css, wsh, sct and hta. The worm then used companion techniques by adding a secondary file next to an existing file, tempting the user to click on the wrong file.[13] For example, a picture named dog.jpg caused the computer to create a new file called dog.jpg.vbs alongside the actual file. For computer users and network managers, this was just the beginning of the I Love You nightmare. Within days of the first attack, the Symantec AntiVirus Research Center identified 29 versions of the worm.

Imagine that you receive an e-mail titled LOVE-LETTER-FOR-YOU with an attachment. A small smile spreads across your face as you see it is from a friend (or not). Your default Windows system does not show the .vbs extension, which may have given you a subtle warning of the impending disaster. You eagerly open the attachment. That is when the worm uses your Microsoft Outlook to send a message to everyone in any address books you have, including global address books that could potentially hold thousands of addresses. If connected to a network, you just became the Typhoid Mary of your organization. When co-workers receive your e-mail and open the attachment, they again send an e-mail to everyone in the organization. The organization servers soon become overloaded and are brought down within minutes.

The story does not end there. In early June, another VBS worm plagued computer networks. Symantec named it "VBS.Stages.A." It was a mass mailer that spread over e-mail in an attachment, always 39,939 bytes, in the form of a Microsoft Scrap Object file (.SHS file). The SHS extension was not visible, even when Windows Explorer properties were changed to show all extensions. The worm was a multiple-application Internet worm, designed to spread using one of four spreading mechanisms—Pirch, Outlook, mIRC and

available mapped drives.[14] The worm sent an e-mail to all contacts in an MS Outlook address book with a LIFE_STAGES.TXT.SHS attachment and a randomly generated 12-string subject. The attachment was titled "Life stages," "Funny" or "Jokes." Some began with "Fw:" or were followed by "text." After it sent the e-mail, the worm immediately deleted copies of the e-mails to ensure no record of its presence.[15]

Disinfecting computer networks of these worms was a fairly simple process involving downloading updated definitions from Norton and McAfee and some manual processes, but it was disruptive and time-consuming. While fixes were being implemented, DOD agencies took steps to mitigate further infection. Several Army installations limited the size of incoming/outgoing attachments and limited distribution lists to 10 or fewer addresses. These positive controls, in addition to working with Norton and McAfee, permitted DOD agencies to limit damage and stall the infection's spread.

These worms and their variants were identified early in May and June, and although quickly contained, were very destructive. I LoveYou may have surpassed Melissa, a macro virus that infected about 70,000 e-mails on 26 March 2000, in both speed and destructiveness. In the end, I Love You was a single worm that infected millions of computers worldwide and caused an estimated $2.6 billion worth of damage.[16]

The lessons learned from these attacks are being institutionalized. Major Army commands and DOD agencies are establishing comprehensive standing operating procedures to interface between planning and information management directorates, redefining the process for disseminating urgent information assurance (IA) messages, organizing information operations (IO) cells, continuing to use IA tools and conducting IA/IO training. DOD also is considering a standardized status reporting system for Army networks and information.[17]

In the private sector, companies are building or revamping disaster recovery plans, ensuring proactive measures are taken to avoid or contain these viruses. Automated and centralized approaches allow information technology departments to restore PCs to their previous configurations simultaneously without physically touching each station.[18] With proactive plans, rapid response and improved technology, we can contain contagion in cyberspace.

NOTES

1. TruSecure Corporation, "2000 Computer Virus Prevalence Survey," online at <www.icsa.net/html/communities/index.shtml>.
2. Ibid.
3. Symantec Corporation, "Computer Viruses: Past, Present and Future," Anti-Virus Research Center, online at <www.symantec.com/avcenter/reference/corpst.htm.>
4. Ibid.
5. Lowenthal, "Overview of Computer Viruses," Information Paper, School of Advanced International Studies - Institute for Advanced Study, March 1999.
6. Ibid.; and additional information concerning virus variations and symptoms is found at the following websites:
<www.rootshell.com> exploits; <www.insecure.org/sploits.html> (exploits);
<http://sourceofkaos.com/homes/madokan/virusindex.html> (virus information);
<http://ciac.llnl.gov/ciac/CIACVirusDatabase.html> (virus information);
<www.snafu.de/~madokan/mvic/viruscont.html> (virus creators);
<www.symantec.com/avcenter/reference/corpst.html> (virus paper);
<http://vil.mcafee.com/villib/alpha.asp> (virus information); and
<www.virusbtn.com> (virus information).
7. Jo Ann Davy, "Virus Protection," *Managing Office Technology* (1998), online at <http://proquest.umi.com.pqweb>.
8. V-One, "'I Love You' Virus Demonstrates Businesses Vulnerability to Electronic Terrorism," *V-One News*, online at <www.v-one.com/news/00-05-05.html>.
9. Deborah Russell and G. T. Gangemi, "Viruses and Other Wildlife," *Computer Security Basics* (1992), 82.
10. DOD Computer Emergency Response Team (DOD-CERT), "VBS/LOVELETTER VBScript WORM," online at <http://coeeng.ncr.disa.mil/diicoe_iava/2000-a-0002.htm>.
11. Kristen Philipkoski, "How the Slimy Worm Works,"*2000 Wired Digital Inc.*, online at <www.wirednews.com/news/technology/0,1282,36129,00.html>..
12. DOD-CERT.
13. Ibid.
14. F-Secure Corporation, "F-Secure Virus Information Pages," online at <www.f-secure.com/v-descs/stages.htm>.
15. DOD-CERT, "Life Stages Worm," online at <www.cert.mil/pub/bulletins/dodcet2000/2000-t-0009.htm>.
16. ON Technology, "ON Technology Delivers Rapid Disaster Recovery as Well as Future Prevention for Love Bug and Other Viruses," online at <www/on.com/presslib/pressre;/on051000,htm>.
17. Assured Information for America's Power Projection Army, "I Love You," Virus Lessons Learned Report, US Army Forces Command, Fort McPherson, Georgia, online at <http://freddie.forscom.army.mil/jwsd/11_report.pdf</jwsd/11%20report.pdf>.
18. ON Technology.

Colonel John C. Deal, US Army, is commander, US Army Information Systems Engineering Command, Fort Huachuca, Arizona. He received an M.S. from the Naval Postgraduate School, and has earned two M.A. degrees from the Naval War College and Salve Regina University. He is a graduate of the US Naval Command and Staff College and the US Army War College Fellowship Program at the University of Pittsburgh. He has served in various command and staff positions, including executive officer, Office of the Director of Information Systems for Command, Control, Communications and Computers (ODISC4), Alexandria, Virginia; staff member of the Secretary of Defense Strategic Studies Group, Washington, DC; and deputy director of Standards, Architecture Directorate, ODISC4.

Major Gerrie A. Gage, US Army, is the operations officer, New Systems Training Office, Directorate of Combat Development, 306th Military Intelligence Battalion, Fort Huachuca. She received a B.S. from Florida Southern College, an M.S. from Florida Institute of Technology, an M.S. from the University of Missouri at Rolla and an M.A. from Webster University. She has served in various command and staff positions, including concepts officer, Maneuver Support Battle Laboratory, Fort Leonard Wood, Missouri; and operations officer, Operational Test and Evaluation Command, Alexandria, Virginia.

Robin Schueneman is a Sytex support contractor to the Information Assurance Directorate, ODISC4. She received a B.A. from the University of North Carolina at Chapel Hill. She has been integral in spreading hacker threat awareness by presenting a "live hack" to the Army's senior leadership. She is currently leads the ODISC4 Information Assurance Vulnerability Alert Compliance Verification Team.