digit

Fast Track *to*
VIRUS PROOF
YOUR PC

Jasubhai
DIGITAL MEDIA

YOUR HANDY GUIDE TO EVERYDAY TECHNOLOGY

Fast Track
to
Virus Proof
Your PC

By Team Digit

Credits

The People Behind This Book

EDITORIAL
Sachin Kalbag Editor
Aditya Kuber Coordinating Editor
Gagan Gupta Writer
Robert Sovereign-Smith Copy Editor
Ram Mohan Rao Copy Editor
Renuka Rane Copy Editor

DESIGN AND LAYOUT
Jayan K Narayanan Lead Designer
Harsho Mohan Chattoraj Illustrator
Vijay Padaya Layout Artist
Sivalal S Layout Artist

April 2005
Free with Digit. Not to be sold separately. If you have paid
separately for this book, please e-mail the editor at
editor@thinkdigit.com along with details of location of
purchase for appropriate action.

Defend Yourself

Perhaps the single most dreaded computing experience is when you find that your PC has been infected with a virus. Your data may be lost forever, and you can't keep your computer from crashing long enough to fix it. More often than not, you end up wasting precious constructive time trying to undo the damage caused.

The sad thing is, a majority of malicious software is written by some of the most brilliant computer minds across the world. Why, is a question best left unasked, as the working of individual human minds is too varied and complicated to comprehend.

This month, *Digit Fast Track* will take you through the world of viruses. You will find eight sections in this book, which will cover everything you need to know about viruses—from what viruses are and who makes them, to how to tell when, or check whether or not you have been infected, to killing the viruses.

Each section will demystify an aspect of viruses that every user should know. We start with the basics in Chapter I—from what a virus is, how it works and the different types of viruses, to other types of malicious software and how to tell whether you have been infected. Chapter Two introduces you to the knights in shining armour, called antivirus applications, and what you should look for when trusting one to defend your PC and your data. Chapter Three will give you a little history about viruses; Chapter Four details the precautions you need to take in order to stay safe from attack, and Chapter Five taken an in depth look at the best security software available today.

There is also a special White Papers section, where you can read some of the best literature ever written on this subject. There is also a Bibliography that recommends some excellent virus-related reading, both for the beginner and the expert, as well as a Glossary that demystifies hundreds of terms that may confuse you when reading about viruses or anti-virus technologies.

We hope this book is comprehensive enough to help the hundreds and thousands of readers to deal with and plan their security, and to keep their precious data safe from the millions of threats "out there".

Contents

Viruses: Know Your Enemy

You need to know your enemies before you can attempt to defeat them. This section will give you an indepth look at what a virus really is, and how they work

You will understand to differenciate between the different types of malicious code, and learn how to tell whether your computer is infected or not.

1.1 What is a Virus?

Life as we know it today would be handicapped without computers. From basic communication, finances and even medical science, computers control just about everything that life in a modern society depends upon. In the ideal world, man would respect such power and work towards bettering it to progression as a civilisation. Unfortunately, that ideal world does not exist, which is why great breakthroughs are often followed by people who are hell-bent on bringing it all down. These people only see weaknesses of an innovation in technology and will go any lengths to exploit it, simply because they can. These are the people who create viruses.

A computer virus can be defined as an executable program that is capable of infecting other computer programs by modifying them to include a copy of itself. Just the way people can spread the common cold by being in contact with other people, a computer virus comes in contact with other programs to 'infect' them. By infecting programs, the virus is capable of spreading through an entire network of computers, infecting every machine that's incapable of protecting itself. While doing so, it could do a world

of damage to your computer, which could cost you dear. The damage could consist of important files destroyed, corrupted data, slowing down of the infected computer, interrupted or unexpected closing of important programs, or it could be any or all of these; and chances are you won't realise that your computer is hit by a virus until it's much too late.

Viruses have grown in number and evolved in nature over the past decade. Before that, it was quite all right to have a basic and even outdated anti-virus program on your computer, which would scan floppies or CDs. That simple task would qualify as protection at the time, but we now need active anti-virus programs, constantly running, checking every file you download or execute.

1.2 How Viruses Work
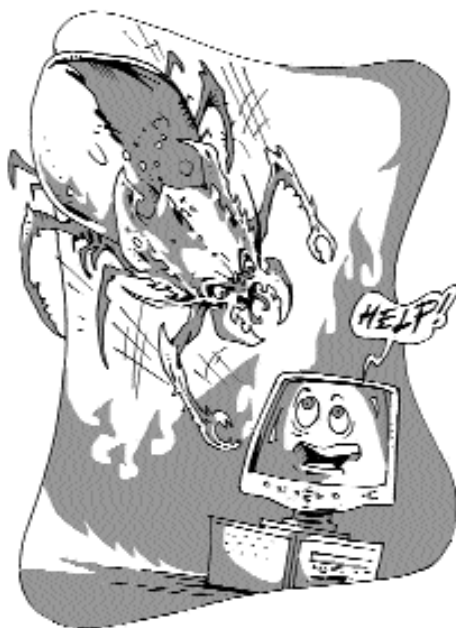
There are tens of thousands of viruses out there, and new ones are discovered every day. It is difficult to come up with a generic explanation of how viruses work, since they all have variations in the way they infect or the way they spread. So instead, we've taken some broad categories that are commonly used to describe various types of virus.

File Viruses (Parasitic Viruses)

File viruses are pieces of code that attach themselves to executable files, driver files or compressed files, and are activated when the host program is run. After activation, the virus may spread itself by attaching itself to other programs in the system, and also carry out the malevolent activity it was programmed for. Most file viruses spread by loading themselves in system memory and looking

for any other programs located on the drive. If it finds one, it modifies the program's code so that it contains and activates the virus the next time it's run. It keeps doing this over and over until it spreads across the system, and possibly to other systems that the infected program may be shared with.

Besides spreading themselves, these viruses also carry some type of destructive constituent that can be activated immediately or by a particular 'trigger'. The trigger could be a specific date, or the number of times the virus has been replicated, or anything equally trivial. Some examples of file viruses are Randex, Meve and MrKlunky.

Boot Sector Viruses
A boot sector virus affects the boot sector of a hard disk, which is a very crucial part. The boot sector is where all information about the drive is stored, along with a program that makes it possible for the operating system to boot up. By inserting its code into the boot sector, a virus guarantees that it loads into memory during every boot sequence.

A boot virus does not affect files; instead, it affects the disks that contain them. Perhaps this is the reason for their downfall. During the days when programs were carried around on floppies, the boot sector viruses used to spread like wildfire. However, with the CD-ROM revolution, it became impossible to infect pre-written data on a CD, which eventually stopped such viruses from spreading.

Though boot viruses still exist, they are rare compared to new-age malicious software. Another reason why they're not so prevalent is that operating systems today protect the boot sector, which makes it difficult for them to thrive. Examples of boot viruses are Polyboot.B and AntiEXE.

Multipartite Viruses

Multipartite viruses are a combination of boot sector viruses and file viruses. These viruses come in through infected media and reside in memory. They then move on to the boot sector of the hard drive. From there, the virus infects executable files on the hard drive and spreads across the system.

There aren't too many multipartite viruses in existence today, but in their heyday, they accounted for some major problems due to their capacity to combine different infection techniques. A significantly famous multipartite virus is Ywinz.

Macro Viruses

Macro viruses infect files that are created using certain applications or programs that contain macros. These include Microsoft Office documents such as Word documents, Excel spreadsheets, PowerPoint presentations, Access databases, and other similar application files such as Corel Draw, AmiPro, etc.

Since macro viruses are written in the language of the application, and not in that of the operating system, they are known to be platform-independent—they can spread between Windows, Mac, and any other system, so long as they're running the required application. With the ever-increasing capabilities of macro languages in applications, and the possibility of infections spreading over networks, these viruses are major threats.

The first macro virus was written for Microsoft Word and was discovered back in August 1995. Today, there are thousands of macro viruses in existence—some examples are Relax, Melissa.A and Bablas.

Network Viruses

This kind of virus is proficient in quickly spreading across a Local Area Network (LAN) or even over the Internet. Usually, it propagates through shared resources, such as shared drives and folders.

Once it infects a new system, it searches for potential targets by searching the network for other vulnerable systems. Once a new vulnerable system is found, the network virus infects the other system, and thus spreads over the network. Some of the most notorious network viruses are Nimda and SQLSlammer.

E-mail Viruses

An e-mail virus could be a form of a macro virus that spreads itself to all the contacts located in the host's email address book. If any of the e-mail recipients open the attachment of the infected mail, It spreads to the new host's address book contacts, and then proceeds to send itself to all those contacts as well. These days, e-mail viruses can infect hosts even if the infected e-mail is previewed in
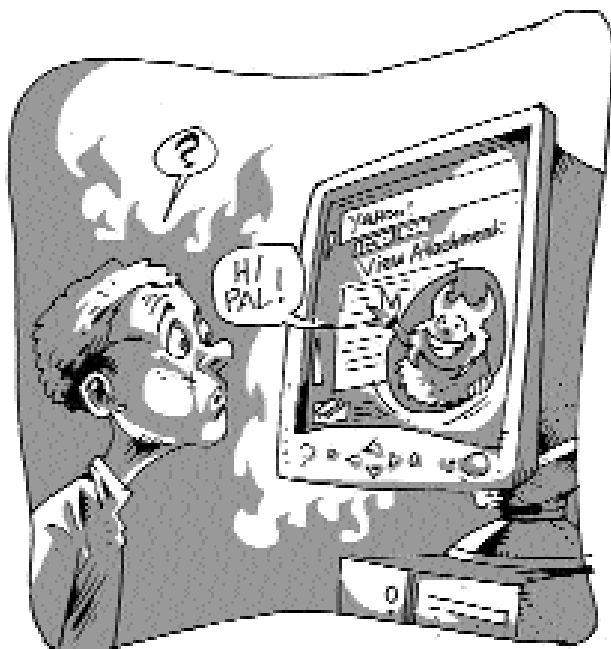
a mail client. One of the most common and destructive e-mail virus is the ILOVEYOU virus.

There are many ways in which a virus can infect or stay dormant on your PC. However, whether active or dormant, it's dangerous to let one loose on your system, and should be dealt with immediately.

1.3   Other Malicious Software

Earlier, the only way a computer was at risk was when you inserted an infected floppy. With the new age of technology, every computer is interconnected to the rest of the world at some point or the other, so it's difficult to pinpoint the source and/or time of the infection. As if that weren't bad enough, new-age computing has also brought about a new breed of malicious software. Today, the term 'virus' has become a generic term used for all the different ways that your computer can be attacked by malicious software. Besides the type of viruses we mentioned in Chapter 1.2, here's a look at some of the newer problems we face today.

Trojan Horses: The biggest difference between a Trojan horse—or Trojan—and a virus is that Trojans don't spread themselves. Trojan horses disguise themselves as useful software available for download on the Internet, and naïve users download and run them only to realise their mistake later.

A Trojan horse is usually divided into two parts—a server and a client. It's the client that is cunningly disguised as important soft-

ware and placed in peer-to-peer file sharing networks, or unofficial download sites. Once the client runs on your system, the attacker—the person running the server—has a high level of control over your system, which can lead to devastating effects depending on the attacker's intentions. Trojan horses have evolved to a tremendous level of sophistication, which makes each one significantly different from the other. We have categorised them roughly into the following:
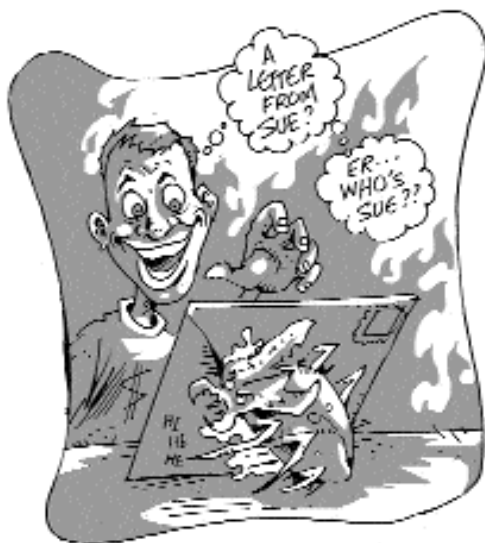
❍ **Remote Access Trojans**: These are the most commonly available Trojans. These give an attacker complete control over the victim's computers. The attacker can go through the files and access any personal information about the user that may be stored in the files, such as credit card numbers, passwords, and important financial documents.

❍ **Password-sending Trojans**: The purpose of such Trojans is to copy all cached passwords and look for other passwords as you enter them, and send them to specific mail address, without the user's knowledge. Passwords for restricted Web sites, messaging services, FTP services and e-mail services come under direct threat with this kind of Trojan.

❍ **Keyloggers**: These log victims' keystrokes and then send the logs to the attacker. The attacker then searches for passwords or other sensitive data in the log files. Most of them come with two functions, such as online and offline recording. Of course, they can be configured to send the log file to a specific e-mail address on a daily basis.

❍ **Destructive**: The only function of these Trojans is to destroy and delete files. They can automatically delete all the core system files on your machine. The Trojan could be

controlled by the attacker or could be programmed to strike like a logic bomb-starting on a specific day or at specific hour.

❍ **Denial of Service (DoS) Attack Trojans**: The main idea behind this kind of Trojan is to generate a lot of Net traffic on the victim's machine, to the extent that the Internet connection is too overloaded to let the user visit a Web site or download anything. Another variation of a DoS Trojan is the mail-bomb Trojan, whose main aim is to infect as many machines as possible and simultaneously attack specific e-mail addresses with random subjects and contents that cannot be filtered.

❍ **Proxy/Wingate Trojans**: These types of Trojan turn the victim's computer into a proxy/wingate server. That way, the infected computer is available to the whole world to be used for anonymous access to various risky Internet services. The attacker can register domains or access pornographic Web sites with stolen credit cards or do similar illegal activities without being traced.

❍ **FTP Trojans**: These trojans are probably the most simple, and are outdated. The only thing they do is open port 21—the port for FTP transfers—and let everyone connect to your machine. Newer versions are password-protected, so only the attacker can connect to your computer.

❍ **Software Detection Killers**: These trojans kill popular antivirus/firewall programs that protect your machine to give the attacker access to the victim's machine.

A trojan could have any one or a combination of the above mentioned functionalities.

Worms: Computer Worms are programs that reproduce and run independently, and travel across network connections. The main difference between viruses and worms is the method in which they reproduce and spread. A virus is dependent upon a host file or boot sector, and the transfer of files between machines to spread, while a worm can run completely independently and spread of its own accord through network connections.

The security threat of worms is equivalent to that of a virus. Worms are capable of doing a whole range of damage such as destroying essential files in your system, slowing it down to a great extent, or even causing some essential programs to crash. Two famous examples of worms are the MS-Blaster and Sasser worms.

Spyware: Spyware is the new-age term for advertising-supported software (Adware). Advertising in shareware products is a way for shareware authors to make money, other than by selling it to the user. There are several large media companies that offer to place banner ads in their products in exchange for a portion of the revenue from banner sales. If the user finds the banners annoying, there is usually an option to get rid of them by paying the licensing fee.
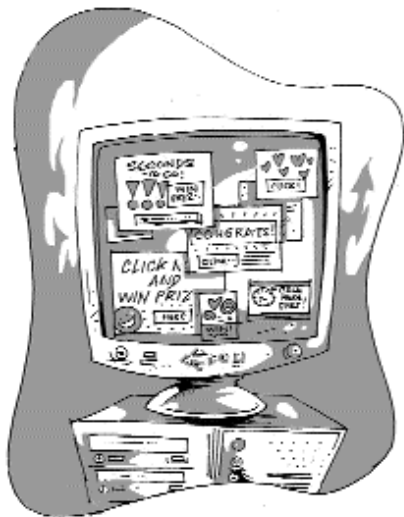
Unfortunately, the advertising companies often also install additional tracking software on your system, which is continuously using your Internet connection to send statistical data back to the advertisers. While the privacy policies of the companies claim there will be no sensitive or identifying data collected from your system and that you shall remain anonymous, the fact

remains that you have a server sitting on your PC that is sending information about you and your surfing habits to a remote location, using your bandwidth.

Spyware has been known to slow down computers with their semi-intensive usage of processing power, bringing up annoying pop-up windows at the most inappropriate times and changing your Internet browsing settings such as your home page or default search engine to their own services.

Even if many do not consider this illegal, it is still is a major security threat, and the fact that there's no way to get rid of them makes them as much of a nuisance as viruses.

Logic Bombs: A logic bomb is a program which has deliberately been written or modified to produce results when certain conditions are met that are unexpected and unauthorised by legitimate users or owners of the software. Logic bombs may reside within standalone programs, or they may be part of worms or viruses. A variation of the logic bomb is the time bomb that 'explodes' at a certain time. An example of a time bomb is the infamous 'Friday the 13th' virus.
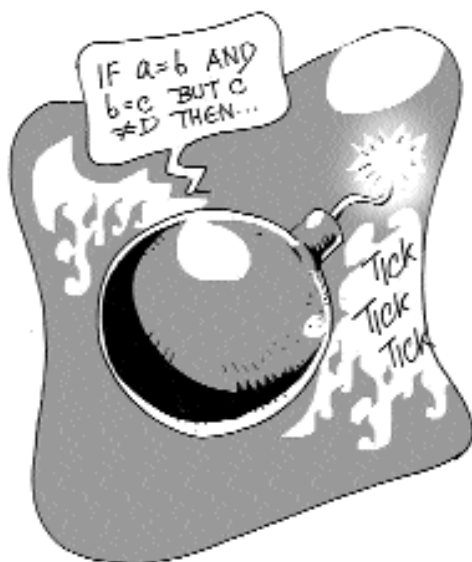
1.4   Avoiding Infection

There's never really a way of being truly protected from viruses, especially when your computer is always connected to some form of network. There are chances that a virus will get to your computer before the resident anti-virus program is even aware of its existence. Still, there are ways to avoid infection by following a set of simple guidelines.

❍ Make sure you have a clean boot CD handy at all times. Your original operating system installation CD should be bootable, so that will do.

❍ If your anti-virus has an option of making a bootable CD, take some time off to make one of those. You will appreciate the effort if the need ever arises.

❍ Use a well-reputed anti-virus software and update it daily.

❍ Make sure your anti-virus automatically scans any newly insert-ed discs for viruses, especially if you tend to exchange data between your office and home computers.

❍ Avoid opening mails with attachments unless you're absolutely sure they are from trusted sources.

❍ If you're using an e-mail client on your computer such as Microsoft Outlook, Outlook Express or Mozilla Thunderbird, disable the message preview pane. This way you can filter the messages you open by the sender's name or the subject line.

❍ Keep all your documents and important programs backed up on a CD, or any other storage media.

Taking these steps won't guarantee that you stay virus-free, but you will surely be more protected and prepared for a virus attack.

1.5   Common Symptoms and Precautions

The problem about virus attacks is that unless your anti-virus tells you, you have no way of being sure that your computer is not infected. Still, there are a few symptoms that you should look out for. These include:

❍ Your computer always stops responding when you try to use certain software. This could also take place due to corruption of an essential file required by that software.

❍ You received an e-mail message that has a strange attachment. When you open the attachment, dialog boxes appear, or a sudden degradation in system performance occurs.

❍ There is a double extension on an attachment that you recently opened, such as .jpg .vbs or .gif. exe.

❍ An anti-virus program is disabled for no reason and it cannot be restarted. The computer may not allow re-installation of the anti-virus.

❍ Strange dialog boxes or message boxes appear on the screen.

❍ Someone tells you that they have recently received e-mail messages from you containing infected attached files, and you are sure you never sent any such mails.

❍ New icons that you did not place on the Desktop appear, and are not associated with any recently installed programs.

❍ Strange sounds or music plays from your speakers unexpectedly.

❍ A program disappears from the computer, and you didn't uninstall it.

❍ Windows will not start because certain critical system files are missing, and you receive error messages listing those files.

❍ The computer starts as expected some of the time, but at other times, stops responding before the desktop icons and taskbar appear.

❍ The computer runs very slowly and it takes a long time to start.

❍ Out-of-memory error messages appear, even though your computer has plenty of RAM.

❍ New programs do not install properly.

❍ Windows restarts unexpectedly.

❍ Programs that used to run now stop responding frequently. If you try to remove and reinstall the software, the issue continues to occur.

❍ A partition completely disappears.

Note that none of the above is a sure-shot sign of a virus infection. There could always be a software glitch, or a loose data cable, or even mere compatibility issues that could be causing such errors. The best thing to do is always keep an anti-virus installed on your computer.

As we mentioned above, there are chances that the erratic behaviour may not even be a virus, especially if it's not detected by the resident anti-virus. Still, to be safe, run a complete virus scan on your computer with the latest anti-virus definitions for the installed anti-virus scanner. If you don't have an anti-virus installed, or think it may not be capable of detecting the virus, you can run an online virus scan from any of the Web sites mentioned in Chapter 2.6.

❍ If a virus is detected, use the steps provided in Chapter 2.5 to get rid of it. In case you feel that you may do more harm than good on your own, and might prefer to have an expert handling the situation, then take the following steps as a precautionary measure:

❍ If your computer is connected to a network, unplug the network cable from your computer.

❍ Switch off the computer. Use the proper shutdown sequence instead of simply switching off the power. Using an infected

computer may simply increase the chances of spreading the virus, or may give the virus ample time to do its damage.

❍ Only switch the computer back on when you are ready to rid the computer of the virus.

❍ Advise users of the other computers on the network to scan their machines, just to make sure that the virus hasn't spread there already.

❍ Make sure that the uninfected computers on the network have some protection against the detected virus. This should be considered top priority.

❍ Do not share CDs or DVDs that were burnt on the infected PC without scanning them for viruses first. The same applies for Zip drives or any other writable media that was connected to the infected computer at some time.

The point of the above exercise is to quarantine the infected computer from the uninfected ones till the virus problem is taken care of.

Viruses Under The Microscope
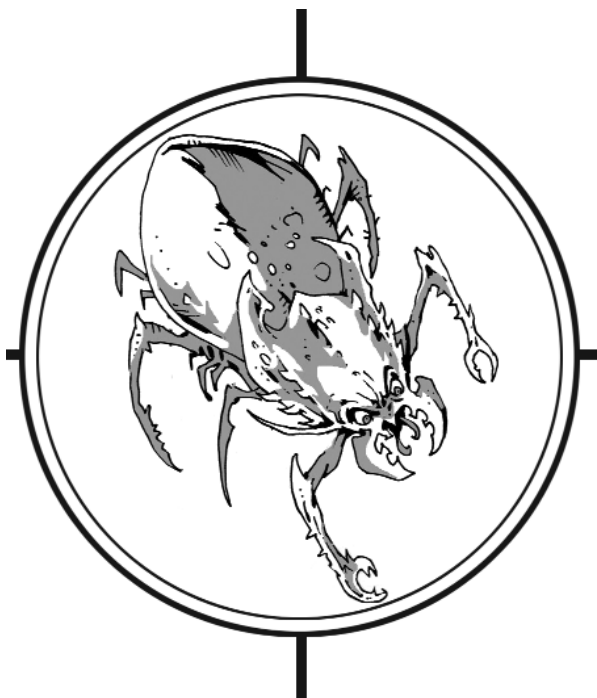
"I have but one lamp by which my feet are guided, and that is the lamp of experience. I know no way of judging the future but by the past."

— Edward Gibbon
British Historian

This section will take you on a trip down virus-memory lane, and also give you an insight into the people who create them.

3.1   A Brief History Of Viruses

In this book we have studied viruses up close and personal, but do we know their entire story? Where did they all begin? What was the purpose behind the creation of the very first virus? What made viruses evolve to this level? For answers to these questions, we need to look at the history computer viruses.

Once Upon a Time...

The when and the where of the very first virus is a little fuzzy to history. The first program which showcased properties of what we now call viruses was called Elk Cloner, a program for the then popular Apple II; this was in 1981. Elk Cloner was quite the fairy-tale character, restricting itself to a fairy harmless rhyme that went like so:

> *It will get on all your disks*
> *It will infiltrate your chips*
> *Yes it's Cloner!*
>
> *It will stick to you like glue*
> *It will modify RAM too*
> *Send in the Cloner!*

Evidently, it was nothing more than a programmer's prank.

Then the world moved on to the mid-'80s; IBM had just created this little thing they called the PC, maybe you have heard of it. It was about the time when viruses started humming tunes of a more destructive nature. One of the first PC viruses discovered was known as the Brain virus, written by two brothers residing in Pakistan. The Brain was a boot-sector virus, infecting 360K floppy disks, but not hard drives. It would occupy unused space on a floppy such that the disk would become useless. Interestingly, Brain was also the first "stealth" virus, hiding itself from detection: if a computer user tried to view the disk sector, Brain would display the original, uninfected boot sector.

Alongside Brain, the 80's saw other major viruses such as "Vienna" and "Cascade". This twosome created something of an epidemic between 1987-89. PC users still remember those days: letters would drop from displays making people draw the obvious conclusion that there was something wrong with their monitors. It wasn't the best of times if your job description involved servicing monitors. But cascading letters weren't the only thing infected computers would display. Some would start playing the "Yankee Doodle" number... weird times, for sure.

With viruses suddenly and irritatingly popping up from the most unlikely of sources, the time was now for saviours. People were desperate for solutions. And so there appeared antidotes. Like the malady, the first cure is hard to pinpoint. Which was the first antivirus, is difficult to identify but it wasn't until 1990 that a visible number of solutions were introduced. And when it rained, it poured. Antivirus solutions were aplenty, including software from IBM, McAfee, Digital Dispatch and Iris. Only a handful of them have survived to this day and even fewer grew from little more than a garage project to major players in the computer security market. Antivirus software had arrived and not a moment too soon.

Come 1990 and viruses started displaying a variety of characteristics. These included Polymorphism—encrypted viruses where the decryption routine code was variable, Armoring—to prevent antivirus researchers from dissembling a virus and Multipartite—able to infect both programs and boot sectors.

The first polymorphic virus was called "Chameleon". By April 1991, everyone was taking shots of "Tequila"—a virus which was Stealth, Polymorphic and Multipartite; a very real and problematic threat. Suddenly, viruses became a lot more threatening.

After Tequila, the idea of a self-encrypting, polymorphic virus gained popularity in the wrong circles and spawned a completely unique software—a polymorphic code generator. Creating slippery viruses was now much simpler.

In early 1992 the famous "Dedicated" virus appeared, based on the first known polymorphic generator, the MtE and the first in a series of MtE-based viruses. The polymorphic generator was essentially an object module (OBJ) file; to get a polymorphic mutant virus from a conventional non-encrypting virus one only needed to link object modules together—the polymorphic OBJ file and the virus OBJ file. The era of "script-kiddies" had dawned and with it an industry of do-it-yourself, 30-minute-viruses. Soon books promising to teach you how to write viruses in weeks made their appearance.

The Michelangelo virus was the first media darling. Much like the Y2K anti-climax, Michelangelo was heralded by much doom and gloom alerts, with predictions of massive, worldwide damages. In actuality, very little happened! The same year the Dark Avenger Mutation Engine (DAME) became the first toolkit that could be used to turn any virus polymorphic. By 1993, polymorphic viruses were populous in virus land. Some celebrities from that era: Bootache, CivilWar, Crusher, Dudley, Fly, Freddy, Ginger, Grog, Haifa, Moctezuma, MVF, Necros, Nukehard, PcFly, Predator, Satanbug, Sandra, Shoker, Todor, Tremor, Trigger and Uruguay.

These viruses required special methods of detection, including emulation of the virus's executable code and mathematical algorithms for restoring parts of the code and data in virus. Polymorphic generators were also proliferating alongside their progeny. Several new ones appeared utilizing complex methods of generating polymorphic code. By the end of 1993 there were four known generators of polymorphic code: MTE 0.90 (Mutation Engine), four versions of TPE (Trident Polymorphic Engine), NED (Nuke Encryption Device) and DAME (Dark Angel's Multiple Encryptor).

Generation Next

As viruses got more prominent, the means to create them kept getting easier and sometime in the middle of 1992 appeared the first do-it-yourself virus kit. July 5, 1992: the first viral code con-

struction set for IBM PC compatibles called VCL (Virus Creation Laboratory) version 1.00 was unleashed. This set allowed generating well commented source texts of viruses in the form or assembly language texts, object modules and infected files themselves.

VCL used a standard WIMP interface—with the help of a menu system one could choose a virus type, the types of files to infect (COM or/and EXE), presence or absence of self-encryption, measures of protection from debugging, inside text strings, plus some 10 additional "effects". Viruses could now use a standard method of infecting a file by adding their body to the end of file, or replace files with their body destroying the original content of a file, or become companion viruses. A virus creator not only had the tool but also choice.

These generator kits kept getting better and the 27th of July saw the first version of PS-MPC (Phalcon/Skism Mass-Produced Code Generator). This set used a configuration file to generate viral source code. The creator file contained description of the virus: the type of infected files (COM or EXE); resident capabilities (unlike VCL, PS-MPC could also produce resident viruses); method of installing the resident copy of the virus; self encryption capabilities; the ability to infect COMMAND.COM and lots of other useful information.

As time went by, virus construction kits got smarter, simpler and more effective. Bad news for the rest of us—practically every teenager with a bad social life and time to spare was churning viruses. Over the years there have been several hundreds of VCL and G2 based viruses and thousands PS-MPC based viruses.

In 1995, Microsoft released the revolutionary Windows 95 and antivirus companies were worried that nobody would need them anymore. The most common viruses were still boot viruses that worked on DOS, but wouldn't replicate on Windows 95. Little did they know... Sometime the same year, macro viruses appeared. These viruses worked in the MS-Word environment. The antivirus industry would keep its job.

The first macro virus went by the name "Concept" and it was pretty unchallenged. Concept soon proliferated to thousands, if not millions of computers in no time it all. Data exchange in the MS Word was now an industry standard, to get infected by Concept, one only needed to open a colleague's file, soon all the documents edited by this newly infected copy of Word would become carriers and the spiral would continue. Adding fuel to this fire was a little thing called the Internet. The reality of infecting frequently used files at the speed of the Internet became one of the most serious problems in computing history.

With time, other macro viruses came to place. In the summer of 1996, there appeared the "Laroux" virus, infecting Microsoft Excel spreadsheets. As it had been with "Concept", these new virus were discovered almost simultaneously by several companies. Of course, tracing the history of viruses, macro virus construction sets soon begun to appear, giving rise to newer and more dangerous kinds of viruses. In the beginning of 1997 came the first polymorphic macro virus for MS Word and the first viruses for Microsoft Office97. The number of macro viruses also increased steadily reaching several hundreds by the summer of 1997. As problematic as macro viruses were—and they are a big problem—macros were not the sole attack vectors.

Threats, Threats Everywhere!

Macro viruses were not the only new threat in 1995. With the rise of the popularity of the Internet, hackers made their presence felt. The Internet gave a lot of opportunities to hackers everywhere. There were all these unsecured servers running important websites and containing vital information just waiting to be hacked. No one was prepared and the hackers took advantage of it. Hacking soon became the next 'cool' thing, with every teenager trying desperately to learn it to impress his friends.

Hackers attacked the Griffith Air Force Base, the Korean Atomic Research Institute, NASA's Goddard Space Flight Center near Washington DC, and its Jet Propulsion Laboratory. Even GE, IBM,

Pipeline and other companies were all hit by the "Internet Liberation Front" on Thanksgiving.

But all was not bad, as these troubled times gave rise to ethical hackers and some of the smartest brains ever to grace the information technology field.

Soon, common vulnerabilities in systems were well known and while the creators of these systems attempted to plug holes, the script kiddies played mischief. Hacking gave rise to Trojan Horses, these new tools didn't require foreknowledge of how systems worked, just the skill to press buttons in the correct sequence. The first Trojan was discovered in 1998 and went by the infamous name "Back Orifice". Back Orifice was a tool that allowed remote administration of any computer that it infected. With Back Orifice, people could take over remote computers, open any files, delete whatever they wished and just about do anything harmful they wished.

While viruses poured in, all was not well in the antivirus camp. In 1997 almost all antivirus vendors were fighting each other in court, or just making a noise about each other. McAfee's "The Number One Choice Worldwide. No Wonder The Doctor's Left Town" led to name calling with its biggest rival, Dr Solomon. McAfee was also in court with Trend Micro over the patent for e-mail data scanning.

Dr Solomon was accused of "cheating", because its scanner supposedly shifted into "Advanced Mode" when it detected a virus, thus enabling it to catch a lot of other viruses that would otherwise be invisible under the "Normal Mode". McAfee claimed that this was the reason that Dr Solomon was fast to scan uninfected disks, and caught more viruses in tests performed with virus collections— though we still fail to see how this was a "bad" feature for an antivirus to have. Symantec was also accusing McAfee of using Symantec code in McAfee products. Needless to say, there was a lot of squabbling.

However, the year ended on a noteworthy event—McAfee

Associates and Network General declared consolidation and Network Associates was born. This company promised antivirus solutions in addition to encryption and network administration services. This was the birth of NAI. Eventually, Dr Solomon was bought by NAI for US $640 million. The event shocked the antivirus world as the conflict between the two antivirus giants was made a thing of the past with a simple bargain. In the process, one of the most notable and technologically strong antivirus software manufacturers lost its identity.

The year 1999 opened a chapter which even today, poses a major threat. Melissa was the first combination of a macro virus and a worm. It used Outlook and Outlook Express to send itself to others via e-mail. Antivirus software scrambled to scan your e-mails, certify virus free e-mails and clog bandwidth with unfortunate and needed overhead. Melissa, of coursed birthed similar threats: the e-mail worms.

Today, the virus threat has increased a level where simple antivirus packages no longer cut it. With Spyware, Worms, Trojans and other malicious software attacking from every medium, we now need more complete security solutions. Indeed, Microsoft has taken security to heart (never too late to start) and is promising to fight the threat of malicious software headlong.

3.2   The Mind Of A Virus Writer

So what makes a person create a virus? What is he trying to prove by spreading chaos and destruction that can only win him jail-time if caught? The popular perception of a virus writer is that of a dysfunctional, pimple-faced teenager; with no girlfriend and no life, who taps out malicious code to a backbeat of trance music. It is a very Hollywood profile and not exactly the most accurate profile. Recent research shows that most virus coders are well-adjusted youths who have normal relationships with their family and friends and intend no real harm with the viruses they write... which is a big problem.

The trouble is they don't believe that their code can actually hurt anyone, which is the mind frame of most teenagers anyways. Some do it out of personal curiosity, to prove a point; others do it to impress their peers, while still others do it to enter the underground computing communities. These communities often consider virus writers as the bottom of their hierarchy and place hackers at the top.

Virus writers represent the wild, unpredictable younger siblings whose unleashed programs are uncontrollable. Hacking on the other hand involves different and more refined skills. A hacker tends to target a specific computing system and make a surgical strike. While hacking is all about gaining control, virus writing is all about uncontrolled mayhem.

Like any adolescent, virus writers tend to mature and change their ways. Most quit the activity once they understand the ramifications of a virus unleashed. Ten years back, virus writers fell in the 14 to 17 year old bracket, while today they're 25 to 28. Women have been known to write viruses. Like all things social, it is difficult to define the mind of a virus writer.

These days the Internet makes it easy to share source code. In the early days of the boot sector viruses, writers needed a certain level of programming skills. Things only got simpler and easier. What's more, virus writers show off their source code at Web sites and distribute the kits we have covered above. Anybody with the inclination can now create a virus.

Society also sends across mixed signals to potential virus creators. While the law seeks to throw them behind bars, security companies have been known to either hire them as consultants or as part of their workforce. Even the press paints them as code cowboys, wild men who live on the edge and dangerously. The attraction is obvious.

So what's the motivation? Malicious intent, honing software skills, exploiting vulnerabilities, experiments, revenge, hobby, peer acceptance, pride... The reasons are as varied as the viruses.

Maintaining Your Vigilance

Not even the best software can save you from disasters if you are not careful about certain things. This section will explain all the necessary precautions you should take, and continue to take, in order to protect your computer from malicious software.

4.1 Patching For Security

The stark reality of life—all software must be patched at some time or another! No, this is not an exaggerated statement. No matter how perfect a software seems when it's released, people are bound to find bugs and security holes in the code. So, to fix these nig-

gling problems, software companies release patches on a regular basis. This really isn't a big deal for offline programs such as your CD burning utility or your image editor, but with any program that functions by connecting to the Internet-an e-mail client, a Web browser or even media players—patching is a priority.
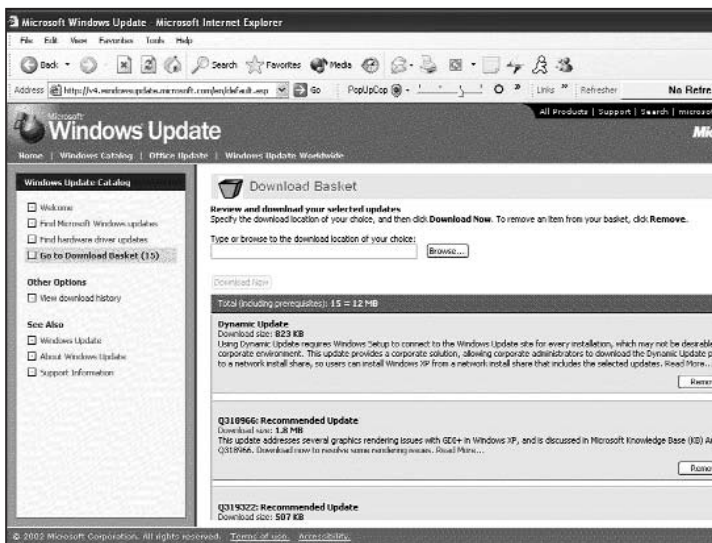
So which software do you need to update? Ideally, you should update all drivers and programs that you use on a regular basis. Your top priority, however, is to keep your operating system updated with the latest patches. Luckily, newer Windows OSes such as Windows 2000 and windows XP, already come with a Windows Update utility that informs you of any new security fixes released for the OS. If you make updation a ritual, your computer will be bet

ter prepared to handle the majority of exploits that come out. The best solution for the Sasser worm, to date, is a Windows update session. Many Linux distribution companies also offer this service.

Software upgradation for the programs you run on a regular basis is also important. A lot of exploits are often found in every-day programs such as browsers and instant messengers, which can leave your computer open to trojans and even hacker attacks. It's best to keep the auto-update option on for these programs, if available, so you don't have to bother checking for updates yourself.

AutoPatcher

For those who can't update their operating systems regularly, there's one application that could be the answer to all your prayers—the AutoPatcher. Available for Windows 2000, Windows XP and Windows 2003, AutoPatcher is a collection of essential patches and updates that have been released for the respective operating systems. Updates include essential patches for the main OS kernel, Internet Explorer, Outlook Express and other Microsoft products. Besides these, it also contains some great freeware to tweak and enhance your computing experience, along with the latest versions of some commonly used utilities.

Autopatcher is also helpful when you do a fresh install of your Windows OS. Instead of running all the patches one after the other manually, or wasting bandwidth connecting to Windows Update and re-downloading all again, Autopatcher can automatically detect the updates needed and patch up your Windows install for you. AutoPatcher is regularly provided with the *Digit* DVD.

4.2 Firewalls And Other Methods Of Protection
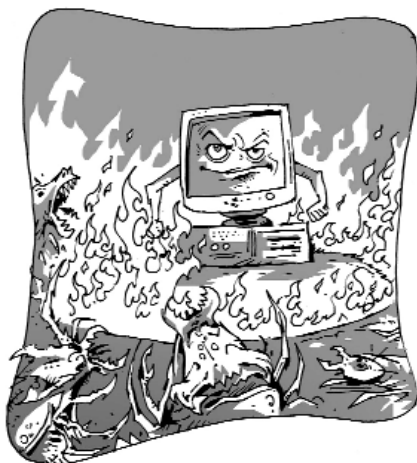
Firewalls were only used as a security method by network administrators to safeguard their servers from unauthorised entry by hackers. The new age of Internet threats and the rise in malicious software means that firewalls are somewhat of a necessity for every computer. Why else would Microsoft include a firewall with its Windows XP SP2 operating system? But before we get into that, let's talk about the basics, starting from what a firewall really is:

In the traditional sense, a firewall is a hardware device or software application that functions in a networked environment to prevent certain communication that's forbidden by security policy. It filters all network packets, and determines whether to allow or block them. It achieves this by screening the requests and determining whether they originate from known and reliable sources. When an unauthorised entry is attempted, say, a hacker trying to access your files, or undetected spyware trying to send out information, the firewall blocks it and also makes your computer invisible to external networks, which is great, as you can't attack what you can't see.
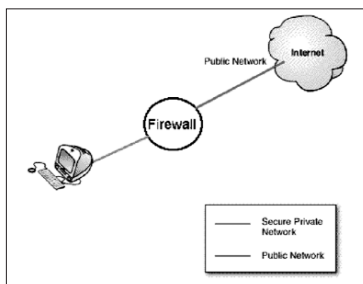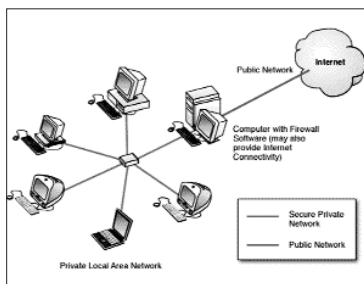
This book will only discuss personal firewalls, which are software applications made for end-users. A personal firewall will not usually protect any more than the one PC it is installed on, unless other PCs are sharing Internet connectivity via the protected PC. There are many misconceptions about firewalls; most people misunderstand what a firewall does for you. Here's a little explanation to clear things up.

What Can A Firewall Do?

Generally, firewalls are configured to protect against unauthenticated logins from the 'outside world'. This, more than anything, helps prevent vandals from logging into machines on your network. More elaborate firewalls block traffic coming into a PC, but allow outgoing traffic. Firewalls can also provide a single 'choke point'—in a situation where a computer system is being attacked by someone dialling in with a modem, a firewall can act as an effective 'phone tap' and tracing tool. Firewalls provide an important logging and auditing function—they provide summaries about the amount and different types of traffic that have passed through it, how many attempts were made to break in, and so on.

What Can't A Firewall Do?

Firewalls can't protect you against attacks that come through software that your firewall isn't protecting. For instance, when you install the firewall and then run a P2P software, the firewall asks whether you want it to monitor that software. More often than not, we disable monitoring of software because we want as little interference as possible. Now if there are security holes in the P2P

software, your computer is as vulnerable to it as when you didn't have a firewall. It's very important to consider what type of software you are disabling firewall protection for.

A major misconception that people have is that firewalls are effective against viruses. An antivirus software can protect your machine from viruses, the maximum a firewall can do is block worm and trojan attacks that originate over the network or via the Internet. However, a firewall that's a part of a security suite, comes bundled with an antivirus that can help your against malicious attacks.

Spam

Spam blockers have become an integral part of a secure computer. There was a time when spam could have been taken as harmless advertising, but now with the constant threat of malicious codes entering your system through spam, you need to take some serious precautions against it. But first, let's start from the top.

Spamming is the buzzword used for the use of any electronic communications medium to send unsolicited messages in bulk. The most common form of spam is delivered in e-mail inboxes as a form of commercial advertising. What differentiates spam from solicited advertising, or newsletters, is that you never signed up to receive it anywhere.

E-mail spamming involves sending identical or almost identical e-mail messages to a large number of recipients. Spam usually contains various tricks to bypass e-mail filters. Spammers obtain e-mail addresses by a number of means—some buy databases from popular Web-based sites that require you to sign up, some harvest addresses from Usenet postings, DNS listings, bulk forwards or Web pages. Some even go to the extent of guessing common names at known domains (also known as a dictionary attack).

Problems Caused By Spam

The reason spam is considered such a major issue among the masses is for a number of reasons. Firstly, we waste a lot of time and bandwidth sorting through and downloading these useless e-mails. Many popular e-mail providers will still deliver fifty to a

hundred spam messages everyday to your inbox, despite their inbuilt spam filtering technology. Not only is it frustrating to filter legitimate mail from the junk, but spam can also fill your allocated inbox space, preventing you from receiving important mails.

It's not only the quantity of spam that's an issue, even the content of the mails are questionable. A large percentage of spam contains ads for pornographic Web sites or other such places that can be unacceptable viewing for many, especially children.

Spam is usually sent from dedicated machines that may not have the required security measures to ensure that the mail is not carrying any malicious code or a virus. Opening a spam mail could immediately infect your PC with spyware or a worm without your knowledge. Another big problem these days are spambots. Some spammers have created various e-mail viruses that will turn your PC into a spambot that will inform the spammer of its existence, and the spammer will command it to send a low volume of spam. This allows spammers to send spam without being caught by their ISPs or being tracked down by anti-spammers as the low volume makes it hard to detect.

Precaution Against Spam

There are many ways to deal with spam, some of them may consume time, or consume money. The oldest method used to get rid of junk mail is manual deletion of everything that comes in your mailbox from an unfamiliar source. It's free, and effective for anyone who gets very little spam in their mailboxes. Unfortunately, not everyone is that lucky when it comes to spam.

A great way to get rid of spam is to manually set filters. Most Web-based e-mail services have a button that allows you to categorise all marked mails as spam, so the next time you get spammed from the same source, the mail is automatically redirected to a junk mail folder. Even e-mail clients have provisions for e-mail filters. You can set up some common spam words as a filter to redirect your mail to the trash or a separate junk mail folder. This method is effective to an extent, but not completely, as many spammers use more and more innovative techniques to get past

the filters, like writing a commonly filtered word like 'Sex' as 'S E X', 'S*E*X' or even '5 E ><'.

The most effective way to avoid unsolicited mail is by using good anti-spam software. Packages such as E-mailProtect (see chapter 5.2) for example, are capable of dynamic, real-time filtering of inbound e-mail, based on approved word and e-mail address lists. It sends all unsolicited and potentially dangerous e-mail to a quarantine folder where you can disable various aspects of the mail in order to preview it safely. Anti-spam software have intelligent engines that have an excellent ability to sort the spam from useful mail, which is perfect for someone who's mailbox is generally flooded with junk. You can easily setup whitelists and blacklists to help the anti-spam utilities identify a useful mail that may come across as spam. The only drawback is that most anti-spam software work only with e-mail clients, so if you're using a Web-based mail facility, an anti-spam application will not be much help.

Privacy Protection

A lot of Internet users don't realise that what they do online can be easily traced back to them. In fact, if you just monitor an Internet user's activity for a period of time, you can tell what's happening in his life. It's a scary thought that if someone takes enough initiative, all your information can be easily accessible like an open book.

To access most services on the Internet, we need to give out some accurate private information, such as our addresses, what we do, how much we earn and even our credit card numbers. Though most Web sites are quite secure with this information, chances are, there are cookies left behind on your machine that store everything you've written. Your browser history stores information about every page you surfed, and some of the pictures that you accessed on the Net can be found in your temporary Internet files.

We don't intend to make you paranoid about technology or the Internet, in fact most home users are not really under much threat at all; but a corporation with many rivals in the same business needs to be careful about the kind of traces it leaves behind.

If you think you could be under a similar threat, you should definitely invest in privacy protection software.

There are many solutions available for privacy protection: some packages are available independently, while others come in a security suite (more in Chapter 5.2). A privacy protection software basically eliminates your Internet and computer usage traces by wiping clean all history, cookies, temporary files etc. depending on what level of protection you set it to. You can clean up all cached files and registry traces at the click of a button, or do it automatically every time you log off your computer. The best part about deleting files using a privacy protection program is that once a file is deleted, it's practically unrecoverable even by a recovery or undelete tool.

4.3   Gadgets Under Threat

If your computer wasn't threatened enough, now you have to con-
tend with malicious code written for your gadgets as well. Virus
makers are moving on to new avenues, and it seems their slogan is
"If it has an OS, we'll
infect it!" It is counter-
productive, can cause
harm to a lot of people,
but sadly, it is reality.

Mobile Phones

The Symbian series 60
phones, such as the
Nokia 6600 and 7610,
have become the
favourite targets for
new age virus makers.
After all, there are
innumerable software
packages available for

these phones that can be downloaded to your computer and installed on the phone—or even installed directly via GPRS. With the platform gaining so much popularity, it was only a matter of time before it captured virus makers' attention. Since the platform is new, people are gullible enough to download whatever upgrades they can find for their phones, and the virus scare hasn't really spread enough to make the people think twice—it's the perfect opportunity for virus makers to spread their work and gain recognition.
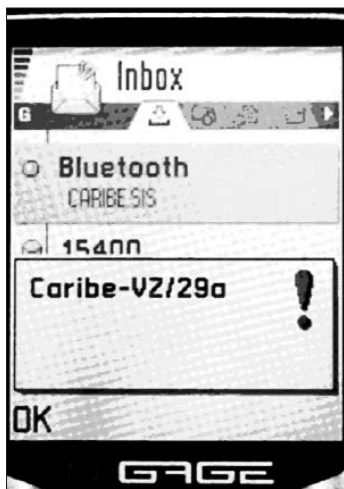
One of the best known viruses for series 60 phones is 'Cabir', a somewhat malicious piece of code that drains batteries and propagates itself via Bluetooth. The fact that a virus as avoidable as Cabir managed to spread amongst the masses, gives us a clue as to how gullible everyday mobile users can be. To get infected by Cabir, your phone needs to be in discoverable Bluetooth mode—visible to all nearby devices. If an infected phone tries to infect your phone, you get a note asking you to accept a message from an unknown device. If you accept, another dialog box asks if you really want to install an unverified program. Then if you click accept, you get a third dialog box that says, "Install Caribe?", which should be sufficient warning to anyone in today's day and age about malicious code. Despite three ominous warnings, people were infected, leaving us to believe that they deserved to be infected in the first place.

The success of Cabir got virus developers thinking, and today, more advanced viruses such as Skulls and METAL Gear for Symbian series 60 phones are posing a threat via Bluetooth. The good thing about Bluetooth is that at least you can reject an invitation sent by the virus; MMS on the other hand is a different ball game. With Bluetooth, viruses can only spread over short ranges,

but with MMS, a virus can send itself across the world—as is the case of CommWarrior.a. Just like a worm on your PC, CommWarrior scans the phone's address book and periodically sends MMS messages to randomly selected con-

tacts. It sends a copy of itself and one of several predefined text messages designed to encourage the recipient to install the application. The good thing is that the application is still only installed by your choice.

There are many other things besides viruses that are considered a nuisance by smartphone users, all related to Bluetooth. A few of these are as listed below:

Bluejacking: Although known to the technical community and early adopters for some time, the process now known as 'Bluejacking' has recently come to focus in the consumer arena, and is becoming a popular mechanism for exchanging anonymous messages in public places. The technique involves abusing the Bluetooth 'pairing' protocol, the system by which Bluetooth devices authenticate each other, to pass a message during the initial 'handshake' phase. This is possible because the 'name' of the initiating Bluetooth device is displayed on the target device as part of the handshake exchange. As the protocol allows a large user defined name field—up to 248 characters—the field itself can be used to pass the message.

**SNARF Attacking**: It is possible, on some mobile phones, to connect via Bluetooth without alerting the owner of the phone. The person connecting to the target phone gains access to the phonebook, messages and other important data. This is normally only possible if the device is in 'discoverable' or 'visible' mode, but there are tools available on the Internet that allow even this safety Net to be bypassed.

Backdoor Attacking: The backdoor attack involves establishing a trust relationship through the 'pairing' mechanism, but ensuring that it no longer appears in the target's register of paired devices. In this way, unless you are actually observing your device at the precise moment a connection is established, you are unlikely to notice anything untoward, and the attacker is free to use any allowed resource for trusted devices, such as file transfers. This means that not only can data be retrieved from the phone, but other services, such as modems or Internet, WAP and GPRS gateways may be accessed without the owner's knowledge or consent.

Bluebug Attacking: The Bluebug attack creates a serial profile connection to the device, thereby giving full access to the AT command set, which can then be exploited using standard off the shelf tools, such as PPP for networking and gnokii for messaging, contact management, diverts and initiating calls. With this facility, it is possible to use the phone to initiate calls to paid numbers, send or read SMS messages, connect to data services such as the Internet, and even monitor conversations in the vicinity of the phone. Monitoring conversations is done via a voice call over the GSM network, so the listening post can be anywhere in the world. Bluetooth access is only required for a few seconds in order to set up the call. Call forwarding diverts can be set up, allowing the owner's incoming calls to be intercepted, either to provide a channel for calls to more expensive destinations, or for theft by impersonation of the victim.

The way the trend is going, it seems that 2005 will see a big rise in the number of mobile phone viruses. That's why companies such as SimWorks International Limited have already worked out an antivirus for the Symbian series 60 phones.

SimWorks Anti-Virus protects phones from all known viruses including Cabir a, Cabir b, Cabir c, the Mosquitoes dialer and the Skulls trojan. The antivirus is basically a
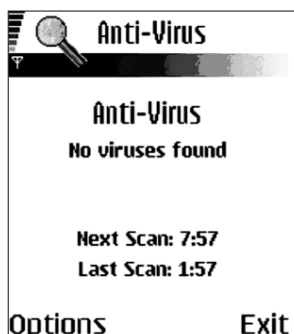
phone based application that auto-starts when you switch on your phone. It scans incoming messages and programs in real-time for viruses, stopping most malicious software at the root itself. Just like a PC antivirus, you also have an option to scan the phone manually or schedule phone scans. You can even download updates from the SimWorks Web site—http://www.simworks. biz/sav/AntiVirus.php

PDA viruses

The world's first PDA virus was discovered running in a Windows CE-based Pocket PC in 2004, but then again, who didn't see that coming? With PDA's getting closer to being full fledged PCs, and the tremendous amount of independent software development for it, it was only a matter of time before virus developers would pick up the SDK and write away.

The first virus discovered on the Pocket PC was a classic Trojan backdoor program called 'Brador.a'. When Brador.a is launched, it copies itself to Windows/StartUp/Svchost.exe so that it starts when Windows starts. In doing so, it continually attempts to send the attacker the IP address of the handheld by e-mail until it succeeds; then it waits for further instructions from the attacker. The virus allows the attacker to remotely list the directory contents, upload

a file, display a message box, download a file and execute specified commands. However, the good thing is that this trojan won't really start acting up until you execute it for the first time, so just be careful about the kind of programs you run on your PDA.
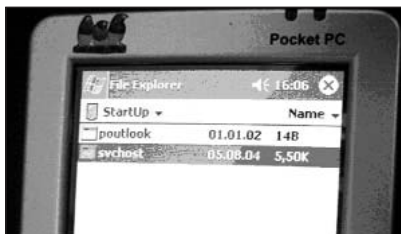
Palm OS owners aren't completely safe either. There have been three major viruses detected for the Palm OS in recent times. First, there's the 'Liberty Crack' Trojan horse program that can wipe out all the files from a PDA running the Palm OS. Then there's also the Palm 'Vapor' virus; another Trojan horse that renders all third-party application icons invisible, appearing as if they had been deleted. The third is the more heinous and more malicious 'Palm OS/Phage' virus, which fills the device's screen with a grey box, crashes the application that is running, and then replicates itself.

As always, prevention is better than cure. As of now viruses on PDAs are not smart enough to spread without the help of user ignorance, so it's up to you to be vigilant about the things you accept, and the applications you install on your PDA. Of course when all else fails there's always the age old antivirus you can depend upon.

Here are a few antivirus packages offered by known companies to keep your PDA secure.

McAfee VirusScan PDA Enterprise 2.0

Platform: Windows Pocket 2002, Windows Mobile 2003

Based on the McAfee scan engine, VirusScan PDA employs advanced detection and cleaning techniques to prevent all kinds of viruses and other malicious code. It features real-time virus detection and on-demand/on schedule detection. The automatic updating ensures that the devices have the latest virus updates.

AirScanner Mobile Antivirus Pro

Platform: PocketPC 2003, Windows Mobile 2003

The AirScanner Mobile Antivirus Pro can quarantine or eradicate embedded viruses and malware with its fast and optimised scanning speeds. Just as any good virus scanner should, AirScanner features automatic, online updates of virus signatures and the scanning engine. In addition to a virus scanner, it also includes powerful tools for debugging Trojan horses with its advanced process discovery tool. With its ActiveGuard feature there's real-time virus scanning by default. For people who value PDA resources at all times can use the slow background scanning option.

Symantec Norton AntiVirus for Handhelds

Platform: Palm OS, PocketPC

From one of the most trusted names in PC antivirus solutions, Norton Antivirus for Handhelds. It has all the required features that can be expected from a PDA antivirus today. The auto-protect feature provides unobtrusive real-time protection against malicious code. Automatic scans can check for viruses after expansion card insertion or desktop synchronisation. On-demand and on-schedule scans allow you to examine applications and files for viruses at any time you want. Virus protection updates are automatically transferred from your desktop computer the next time you synchronise your PDA. Of course, there's also an auto-update feature that automatically downloads the new virus definitions.

4.4   Safe Computing

There are many precautions you can take while computing. Here are a few you should follow if you're paranoid about your security:

Firewalls And Security

❍ Use a firewall if you have an always on Internet connection. Corporations can invest in hardware-based firewall security for their networks, while home users can opt for software-based personal firewalls.

❍ Turn off file sharing on your PC when you don't need it. If a port scan is done on your computer, a hacker may find a back door to your machine and have access to your files via file sharing.

❍ Don't open attachments when you receive e-mail from unknown sources and the subject line seems unfamiliar.

❍ Don't let other people use your computer, unless you really trust them. If you really have to share, then create a guest login for the other users with limited rights.

❍ Routinely update Windows software. The updates will fix many bugs and known security holes within the Windows operating system.

❍ In case, regularly used programs have options for auto-update, keep them on. You never know when a lethal security bug may be discovered.

Privacy Protection

❍ Your account is only as secure as its password. Create passwords with nonsensical combinations of upper and lower case letters, numbers and symbols. Also, change your passwords often. If you must write down or record your password, take steps to disguise the information so no one else can make out what it is.

❍ Look at the privacy policy of the online services you use and also before you enter private information in online forms. However, if you are not satisfied with the policy, or if there is no policy posted, avoid giving any personal information to that site.

❍ Check your browser's cookie settings. You may accept or reject all cookies, or you may allow only those cookies generated by the Web sites you visit often.

❍ Do not provide sensitive personal information such as phone numbers, passwords, addresses, credit card numbers or date of birth in chat rooms, instant messengers, forum postings, e-mails or in your online biography.

❍ Ask yourself if you want an employer, family member, or a marketer to be able to link you to your public postings made in forums, guestbooks, or newsgroups. Most of these services never delete your postings. Even the one's that have disappeared off the Web site are usually accessible in the archives.

❍ Use a pseudonym and a non-descriptive e-mail address when you participate in public forums. Consider obtaining an e-mail address from one of the free Web-based e-mail services for this purpose.

❍ Be aware of the possible social dangers of being online. Harassment, stalking, being 'flamed' (emotional verbal attacks), or 'spamming' (being sent unsolicited messages) are just a few examples. Women can be vulnerable if their e-mail addresses are recognisable as a woman's name. Consider using gender-neutral e-mail addresses and nicknames.

❍ If your children use the Internet, teach them appropriate online privacy behaviour. Caution them against revealing information about themselves and your family.

❍ Use only secure Web sites when you transmit sensitive personal information over the Internet. When you provide your credit card account number to a shopping site, for example, be sure that the transmission is secure. Look for the unbroken padlock at the bottom right of the screen. Also make sure the Web address has the letter 's' after http in the address bar at the top of the page.

❍ Be aware that online activities leave electronic footprints for others to see. Your own ISP can determine what search engine terms you use, what Web sites you visit, and the dates, times,

and durations of your online sessions. Web site operators can

often track the activities you engage in by placing 'cookies' on your computer.

Mobile Phones And PDAs

❍ Frequently back up all data such as your phonebook, calendar, and others to your PC. If you aren't provided with a good back-up software, it's advisable to purchase one.

❍ Don't forget to keep your Bluetooth off when not in use. Even when you turn it on, keep your Bluetooth's visibility mode as hidden from other devices.

❍ Never accept any Bluetooth input from an unknown source. You could get infected with a virus.

Virus Myths And Virus Slayers
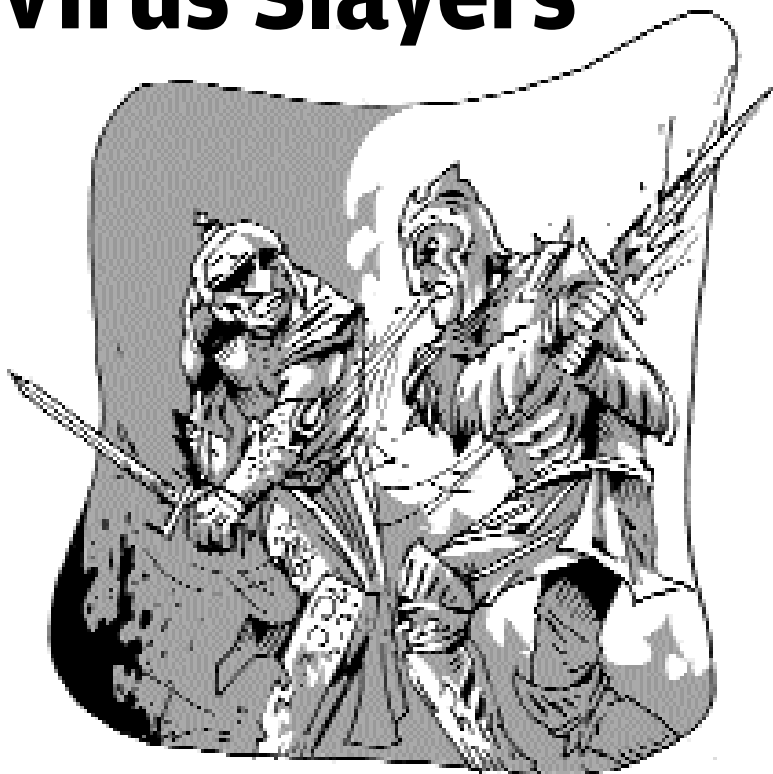
The problem with epic battles is that there is always a few stories to be told, and not all of them are entirely true.

This section will uncover some of these myths and also take a look at 10 solutions that you can entrust your data security to.

5.1   Myths About Computer Viruses

As we have said before, all the suspicious activity that happens on your computer cannot be blamed on viruses. There could be many reasons why your PC has been acting up, such as applications clashing, missing files, hardware incompatibility and much more. There are a lot of myths and false notions that people seem to have towards viruses and their solutions. This section hopes to shed light on some common misconceptions.

Only Microsoft operating systems get viruses

Most virus writers want their work to be famous and get world-wide recognition. With the majority of the world using Microsoft operating systems, there are no prizes for guessing why most malicious code is written for them. There are viruses for Macintosh and Linux computers as well, though not as many.

If the EXE attachment is from someone I know, surely it's safe

Just because you know the person, doesn't mean that the person is aware of the mails he's forwarding. He could unknowingly be forwarding an infected attachment without realizing it, or it could be a worm on his computer spreading its code to everyone on his address book.

The worm can't hurt me if I don't open the attachment

We only wish it was that simple. Today, most worms can infect your computer and then spread to all your contacts even if you just open the mail that contains the worm. You need to pay attention to the subject line of the message.

Installing an antivirus guarantees my protection

Absolutely not! Depending on the kind of antivirus package you have, it may have its own strengths and weaknesses. With viruses getting smarter by the day, there's a strong chance that the latest batch may be undetectable for the antivirus' heuristics system.

I don't use an e-mail client, so I'm safe from e-mail viruses

Though this may be true to a small extent, it's not a foregone con-clusion. Yes, using an e-mail client does give a virus the opportu-

nity to use the security holes of the application, but Web-based e-mail doesn't really have any other advantages. You will still receive viruses in your Web-based e-mail, and they will infect your computer if you download the attachments.

Formatting a hard drive is the best way to get rid of a virus.

Yes, but only if you are not saving any of the data on it. If you plan to backup your data, chances are the virus will embed itself in that backed up0 data, and as soon as you restore the backup, you are infected again! The best ways to get rid of viruses have been explained in chapter 2.5.

5.2   Ten Antivirus Solutions

It can be quite a daunting task to find the perfect antivirus. There are so many options available, and all seem to offer similar features. What you need to do is identify only the features that you need. You also need to keep your system configuration in mind, and remember that an antivirus application is always running in the background-an antivirus application that is system heavy could kill a low-end system.

To help you make your choice easily, we have put together a list of ten of the best antivirus packages across the world. We will tell you each solution's advantages and disadvantages, and what features they offer, and this will help you make a thoughtful buying decision.

So many options...

There are so many customised security solutions available for just about every kind of user today. While the paranoid few may like to pile up on a number of applications, just to be sure that their system is secure from every threat in existence, some are more than satisfied with a barebones antivirus scanner. We have highlighted some antivirus packages that may be ideal for your needs, and also separated them into different categories.

The Complete Security Solution

When you're using your PC for commercial purposes, the data on your machine is usually extremely sensitive and possibly even confidential. You don't want anything happening to it by a virus attack, and you definitely don't want it falling into the wrong hands. For security of this level you need a lot more than a mere antivirus; you need a specialised suite of utilities that secure your machine from most threats. That's exactly where the following software come in.

ZoneAlarm Security Suite 5.5

It's a given that ZoneAlarm is everyone's first choice when it comes to personal firewalls. Now imagine a well reputed brand like that bringing you a complete security system-a package that claims to be the only thing you will require to keep yourself free from viruses and other malicious software attacks. With big names already in the market with similar packages, how does this one hold up? Pretty well actually!

The antivirus module is licensed from Computer Associates. Just like any decent antivirus, this one too is capable of scanning files in real-time as well as on demand and on schedule. With a smart heuristics system in place, the antivirus is effective against all kinds of malicious code, including viruses and trojans, even before there's a signature available for them. It works out to be extremely effective in conjunction with the firewall.

The firewall module was already perfected by Zone Labs, so it's pretty obvious that it's the firewall that's the biggest selling point of this package. The firewall features intrusion blocking technology that systematically identifies hackers and blocks access attempts, along with a stealth mode that automatically makes your computer invisible to anyone on the Internet. The firewall is pretty easy to set up and configure, which makes it convenient even for novice users.

The antispam module is licensed from MailFrontier. It integrates well with e-mail clients such as Outlook Express and filters out all the incoming spam quite effectively. It even monitors the outgoing mail for any suspicious activities, such as too many receivers for a single mail or too many mails being sent in too short a time interval. The IMSecure module protects your Internet messengers from spim and any other suspicious activity.

Other features include modules for privacy protection, which includes cookie control, ad blocking and protection form malicious scripts. There's also a parental lock for preventing access to sites unsuitable for children.

Pros: Excellent firewall; antivirus and antispam.
Cons: antispam doesn't filter your existing inbox.

Panda Platinum Internet Security 2005
Panda Platinum Internet Security 2005 is designed to be the only package you will need to protect yourself from all kinds of problems that you can get into while online. The package integrates an antivirus application, a spam filter, a firewall, and even parental control technology that can block objectionable content such a pornographic sites.

The antivirus is based on Panda's TruPrevent technology, which works on an advanced heuristics engine that can block out most malicious code on its own. That, combined with the constant updates in the form of virus signatures from Panda, makes the antivirus a very strong contender. As with all good antivirus software, this one also has a real-time scanner that checks files on access and a system scanner for on demand and scheduled scans. The antivirus is highly proficient in detecting and cleaning spyware, in fact it's known to be one of the best spyware cleaners in an antivirus package.

However, the firewall module provided by Sygate leaves a lot to be desired. Though the firewall is pretty good at keeping your computer invisible to prying eyes, it's definitely not the most secure option out there. A regular trojan attack would be sufficient to humble the firewall's defences.

The antispam module integrates well with common e-mail clients such as Outlook Express and Eudora. Unfortunately, this too is not really as secure as we would like it to be. It lacks features such as intelligent mail sorting.

Pros: Feature Packed; great antivirus and anti-spyware.

Cons: Antispam not very effective; firewall needs to be more secure.

Trend Micro PC-cillin Internet Security 2005

PC-cillin Internet Security 2005 includes an antivirus, a full-fledged firewall and also spyware and spam blocking tools. Just as the name suggests, it's an overall solution to all kinds of threats you can face while surfing the Internet.

First up, the antivirus system is commendable for its silent and stealthy performance. With absolutely no drop in system performance, you'll hardly even notice that it's there. Even when scanning disks, you can continue with basic tasks without noticing a drop in system performance. Also, its feature set is par for the course with real-time as well as on demand and on schedule scanning.

Though this suite does not scan for spyware in real-time, it automatically scans for spyware as well, when running a virus scan. You can also run the spyware scanner separately. When PC-cillin detects potential spyware, it gives you the option of deleting

the program or going to Trend Micro's support site to learn more about it. The only problem is that the spyware scanner is a little paranoid, and even detects regular adware programs as spyware, which may mean that a few adware-supported programs you like will stop working after a spyware cleansing.

Trend Micro's spam filter tags any e-mail with objectionable content with the word "SPAM" in the subject line. This allows for easy rule filtering and prevents potentially important e-mail from being deleted automatically by the application. The spam feature also includes customisable "whitelists" and "blacklists" and the ability to submit improperly tagged e-mail directly to the analysts at Trend Micro.

The personal firewall comes with selectable profiles that best match your use. It has the ability to automatically switch profiles when you change networks, which is extremely handy for laptop users. The firewall silences ports, keeping you virtually invisible on the Internet, and also controls network traffic to stop worms from spreading.

Pros: Light on the system; great interface.
Cons: No real-time spyware detection.

Security for a Home User
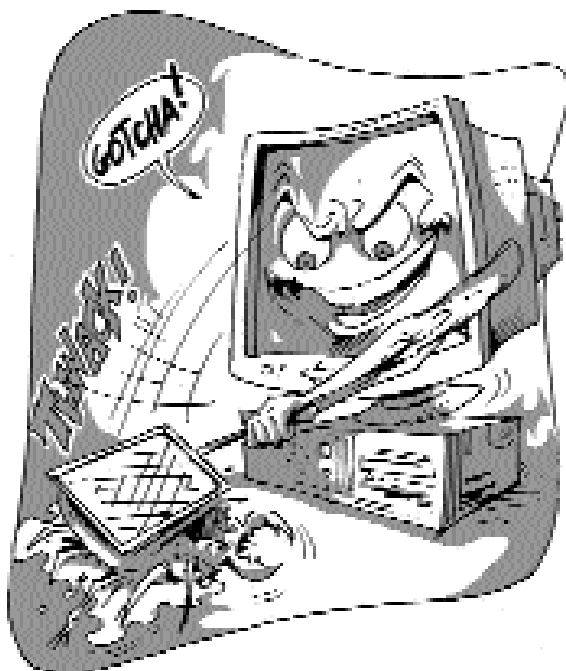
Home users are generally more casual when it comes to security threats than corporations. Firstly, home users don't like spending too much for suites that probably wouldn't be any more effective than a good antivirus, for the level of security threats they face. Many home users prefer to use their PC for entertainment purposes, so having a league of security related software running in the background is not an option, especially when you are trying to eke out maximum performance from your DVD player, or when playing a game. You need something light and effective

Norton Antivirus 2005

Over the years, Symantec's antivirus services have proved to be the first place that one looks to find information on the latest virus outbreaks. Norton Antivirus is one of the most highly used commercial antivirus packages in India, mainly due to its ease of use and availability. Norton Antivirus 2005 is the latest in Symantec's desktop antivirus solutions.

Norton AntiVirus 2004 had a host of missing features, which Symantec have remedied in the 2005 edition. The most notable is an integrated lightweight firewall titled "Norton Internet Worm Protection" that blocks out any unwanted incoming traffic. Though it is effective in most cases, it doesn't monitor any outbound traffic, which makes it less effective as compared to a good standalone firewall.

Like its predecessor, Norton Antivirus 2005 is a robust solution against viruses and also against other Net nuisances such as spyware, adware, trojans, etc. Oddly though, the Norton real-time

scanner doesn't seem to look for spyware, and you can only detect spyware when performing a system scan.

On the downside, Norton Antvirus is still quite system heavy, which may put off a lot of performance enthusiasts. Don't expect to do much on your computer while running a system scan as that will easily drain most of your resources, making every other application painfully slow. Nonetheless, Symantec's reputation with virus definitions, the extensive protection against all kinds of malicious software and the best interface among all antivirus packages, makes Norton a strong option.

Pros: Excellent interface; great virus definition support
Cons: Significantly system heavy; no real-time spyware detection.

McAfee VirusScan Home 9.0

McAfee antivirus packages have always been a great option, but if there was one particular flaw that made some users shy away from it, it was the complicated interface. But version 9.0's tabular interface makes sure that all the required options are well sorted and easy-to-find.

McAfee is generally light on the system as it secretly keeps a watch on all the files accessed by your system. Just like Norton Antivirus 2005, it can detect viruses, trojans and worms in real-time, but detects spyware and adware only during the system scan. The good thing is running a system scan on regular intervals is not really a big issue, because even though the system scanning process
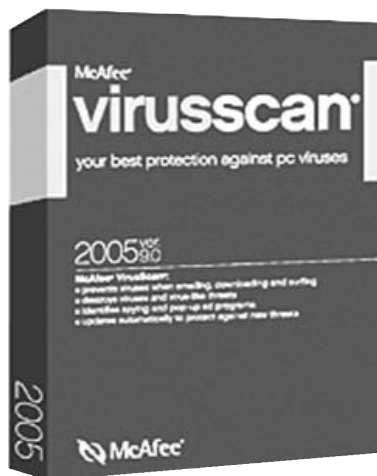
takes longer than other antivirus packages, it uses up a lot less resources, which means that you can continue your work while the system is being checked.

One important feature missing from VirusScan 9.0 is a firewall or at least a port-blocking technology to stop unsolicited inbound packets. These days having a firewall on your system is a must, especially with the colossal rise in malicious software threats. Many other packages have spotted this danger and now come with inbuilt firewalls. If you already have a sturdy firewall running in your system and only need a good antivirus package that doesn't take its toll on your resources, then McAfee VirusScan Home 9.0 is definitely worth some consideration.

Pros: Improved interface; not too heavy on system resources.
Cons: No integrated firewall; no real-time spyware detection.

NOD32 Antivirus System 2.0

NOD32 is a multi-award winning antivirus package that's slowly gaining popularity as the antivirus to have. There are just too many strong features in NOD32 to sideline it while making a decision to buy an antivirus.

The NOD32 Antivirus System is essentially divided into four parts:

❍ NOD32 is the complete on demand and on schedule system scanner.

❍ AMON is a real time scanner that checks files on access.

❍ DMON protects the system from viruses contained in various documents such as macro viruses in Microsoft Word documents.

❍ IMON scans incoming POP3 and HTTP streams protecting you from Web and e-mail threats.
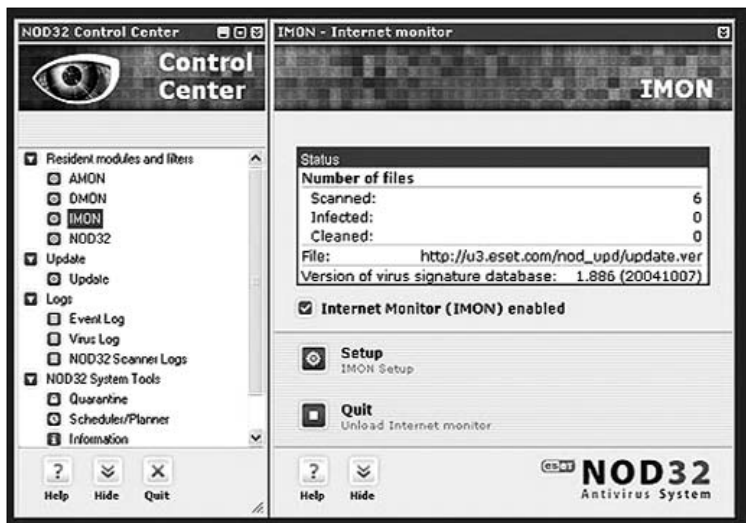
One feature that most advanced users enjoy is its advanced tweaking capabilities. However, casual users may be turned off from the not-so-friendly interface. NOD32 is an ideal solution for gamers who require every ounce of their system resources while playing a game. There isn't a noticeable drop in system resources, which makes it quite alright to keep the antivirus program running while playing games.

The lack of firewall can be considered a drawback, but the heuristics system in NOD32 is one of the best in the business. Even though the virus definition updates keep coming at almost a daily basis, the heuristics system alone can block out any malicious software that's about to act up, making it a valuable antivirus even when it's not completely patched up to existing standards.

NOD 32 protects against all kinds of malicious codes, namely viruses, trojans, worms, spyware and adware.

Pros: Very light on the system; excellent heuristics engine; frequent virus definition updates in small packages.
Cons: No integrated firewall; interface aimed at advanced users.

Avast! 4 Home Edition

Many home users would think twice about spending for an antivirus package. After all, paying a decent sum on a yearly basis to protect data that may not be as important to them isn't exactly a very good idea. That's where the beauty of freeware comes in. Yes, Avast! is free for home usage and you can check it out on the *Digit* DVD.

While freeware, to most people, usually translates to an amateurishly made home project by a pimply faced teenager, we assure you that's not the case here. Avast! antivirus features outstanding malware detection abilities, together with high performance. The interface is simple and easy to follow for most purposes. There's also an option to skin the interface just in case you want to give your antivirus a new look. Why you would want to do that to an antivirus, however, is beyond us.

The antivirus consists of a real-time scanner that scans files for viruses on access, on demand or on schedule. There's also a real-time scanner for your incoming e-mail, which prevents e-mail-based viruses and worms before they get a chance to act up. However, the e-mail functionality is offered for Microsoft Outlook only.

The virus scanner is based on an intelligent heuristic system that detects viruses on your computer and your e-mails. Virus definitions are regularly updated and Avast! intelligently connects to the server and automatically downloads the updates (which are generally quite tiny) whenever you connect to the Internet.

Pros: Free; light on the system and very effective against viruses.
Cons: Barebones antivirus with no frills such as a firewall or spyware scanner.

Specialised Solutions

Though most suites are quite proficient when it comes to moni-toring all kinds of malicious threats, there may be some aspects when they're not exactly performing at a level that they should be. For example, most antivirus solutions have real-time scanners for all kinds of malicious software, but not for spyware, which hap-pens to be a pretty major threat today. For that you may require a specialised software that can secure your PC from these resource hungry programs. The same logic can be applied to some of the other security issues that could be a bit out of the league for your resident antivirus.

EmailProtect

EmailProtect is a dedicated spam-filtering program that doesn't use the simple whitelists and blacklists that other programs depend on. Instead, you fine-tune categories and preferences to teach EmailProtect how to do the work for you. EmailProtect relies on your settings to filter your inbox. You choose the keywords, e-mail address, domain names, and even servers to watch for, and the filter takes over. You can opt to allow or block using over 20 categories (such as Adult, Drugs/Alcohol, Shopping, or Pornography). Categories are pre-defined, which means you don't have to take time to decide details for each, you just pick which you want allowed or blocked.

EmailProtect integrates into your e-mail client and adds a quarantine area. If a message is suspect, the program will send it to quarantine and also notify you immediately, if that's what you want. From there, you can move the e-mail back to your inbox, delete it, or use it to create new filtering rules. The tools are simple to use and everything you need is just a button click away.

EmailProtect features a protected way to preview e-mails with images. You can turn image display on or off quickly. You can also
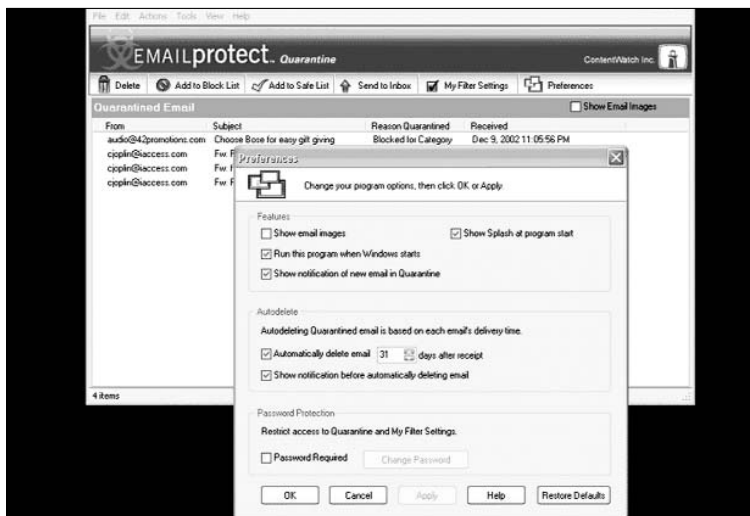
preview spam text without tipping off the sender. Most spam contains "bugs" that send transmissions back to notify the sender that you have opened the e-mail, but the EmailProtect spam view screen prevents this transmission.

Pros: Goes beyond using white lists and black lists; easy to use.
Cons: Needs some manual configuring.

Spyware Eliminator 4.0

When it comes to spyware constantly eating up your computer resources and bandwidth, there's no such thing as being too safe. Spyware Eliminator does exactly what its name suggests-eliminates spyware. The best thing about having a dedicated anti-spyware solution is that it gives you real-time tracking and blocking capabilities. That way, you prevent rather than cure, and spyware is blocked at its source.

Spyware Eliminator 4.0 also offers 'Consumerware'—a new section that separates legitimate Adware companies from the actual spyware so you know what to delete, uninstall or keep.

It's easy enough for casual users to simply install it and then forget about spyware, but at the same time it features some serious customisation options for advanced users.

Pros: Real-time spyware scanning; Consumerware section.
Cons: None

Wormguard 3

Wormguard is arguably the best protection you can have against Internet worms. When your firewall and antivirus fail to detect a worm infecting your computer, that's where specialised software such as wormguard can come to the rescue. Its highly intelligent heuristics system analysis files generically rather than relying on signatures for known worms.

Wormguard provides real-time file scanning on all executed files to ensure they're not infected before the worm even gets a chance to act up. It also neutralises many severe Windows vulnerabilities, such as the use of hidden extensions, multiple file extensions, and excessive spaces in filenames.

The on demand scanner provides Deep-Scanning to detect password-stealers, keystroke-loggers, IRC worms, references to known worm authors, etc.

Pros: Easy to use; highly effective.
Cons: Slightly outdated but new version coming soon.

In-depth

I f you've gotten this far, you've probably learnt a lot already. However, for those of you who like to go into the nitty-gritty of things, the whitepapers that follow should further improve your understanding of the topic at hand. Here, you'll find in-depth material on spyware, adware, the need for a secure operating systems, network security best practices, and so on. And if you thought it's only Windows that's affected by viruses, there's also a paper on viruses that attack Unix systems.

I. Spyware And Adware

Spyware—software installed on a computer usually without the user's knowledge or permission—along with adware and other similar software, gathers information and sends it back to the advertiser who initiated it or other interested parties. Spyware can collect and transmit information such as keystrokes, Web surfing habits, passwords, e-mail addresses, and other sensitive information you may not want to share outside your organisation. Spyware also misuses system resources and bandwidth as it tracks and transmits information. More seriously, spyware can also pose grave security, confidentiality, and compliance risks.

Spyware programs collect data on users and their computing behaviours and then transmit that information back to the spyware host server. These programs can also monitor keystrokes, scan files on hard drives, secretly install other programs, and even make changes to default computer settings. Spyware is often acquired surreptitiously when users download a 'real' application or file, visit certain Web sites, or click on a deceptive pop-up window. Unlike spyware, which is acquired without user knowledge or approval, adware is installed with permission, usually after the user agrees to the terms of a long and confusing End User License Agreement (EULA). These more benign programs also collect information about users or user habits, but typically use it to tailor future pop-up advertisements to users' preferences for marketing purposes. These programs cause performance problems and use expensive computing resources—processing power, drive space, and bandwidth. They can also cause software conflicts with legitimate programs and affect employee productivity. Of most concern to organisations, however, is the fact that spyware compromises information security and consumes valuable IT Help Desk resources. Organisations whose investors and clients rely on them to safeguard personal, medical, and financial information need a

way to prevent spyware from covertly accessing and transmitting critical corporate information. Similarly, organisations whose Help Desk resources are burdened to correct, often by 're-imaging' entire systems thereby preventing the corruption of the desktop computing environments. The security measures currently in place in most organisations-a combination of a firewall, antivirus software, and spyware/adware removal programs-do not adequately address the threat of spyware. Since firewalls operate at the boundary of the network, they have no visibility into the spyware running inside the network. Anti-virus solutions are not adequate either, since anti-virus software typically doesn't include spyware signatures and cannot prevent spyware from transmitting information. And spyware removal programs, which are targeted to individual consumers not organisations, do not provide a centrally managed solution and do not adequately address the burden of application conflicts. Organisations need a way to keep spyware from gaining access to their systems in the first place. To do this, organisations must be able to prevent employees from visiting sites that distribute spyware and from downloading applications that are infected with spyware. For spyware that may be brought on to the desktops through other channels, such as home or mobile laptop use, or via CDs or eFlash drives, organisations also need a way to stop spyware from ever launching, thereby protecting the corruption of that desktop, as well as preventing the transmission of data back to host servers.

Some spyware programs collect information using 'keystroke loggers', which capture information about the user's computer activities, including cookies and time spent on certain sites. Some capture all keystrokes users make; others are more focused, recording Web sites visited, passwords, e-mails, credit card numbers, and so on. Most keyloggers are invisible and save recorded keystrokes into a log file that is transmitted periodically back to the host server. Some can even record both sides of instant messaging chat conversations (for example, MSN Messenger and Yahoo! Messenger).

Spyware can also read a computer's unique hardware ID num-

ber (MAC address) and IP address, and can combine that information with surfing habits and correlate it with any personal information provided during a 'free' software download or when a file attachment was opened. This information can then be traded with affiliate advertisers, building a complex dossier on individual users and what they like to do on the Internet.

Other programs are simple, 'useful' applications such as clocks, calendars, or mouse pointers, which are attractive bait for downloading spyware.

Although similar, adware is distinguished from spyware by the fact that, when downloading adware, the user is first given an opportunity to agree to its being placed on his or her computer. The explanation of an adware program and what it will do is often buried in a long, complex EULA that many users simply scroll through and accept without reading completely. In practice, adware acts as spyware. Both may trigger the display of pop-up or banner advertisements, and both may gather and transmit information from the user's computer.

How Spyware And Adware Can Be Acquired

❍ **When users unknowingly give their permission while downloading or installing applications**: Before installing most software programs, users are required to read and sign an End User License Agreement. But EULAs are long, confusing, and sometimes even deceptive. From a legal standpoint, everything may be duly disclosed in the EULA, but EULAs are often so long and complex that many users just click through them, never stopping to read them closely.

❍ Another method bypasses the security settings altogether by exploiting a bug in Internet Explorer versions 4 and 5. These versions allow Web scripts to gain access to a hard drive by overflowing the browser with data. Malicious Webmasters use this exploit to install spyware or modify the way the browser works.

❍ By simply visiting certain Web sites: Some spyware is secretly downloaded when a user launches a program acquired from a Web site. For example, a pop-up may notify the user that a spe-

cial plug-in is required to run a video or movie file. In this case, what appears to be a legitimate plug-in could actually be spyware. Some spyware takes advantage of known vulnerabilities

in the Microsoft Windows operating system and Internet Explorer browser to secretly place spyware onto the user's computer. For example, one such method involves pushing malicious JavaScript and VBScript code to the user's Web browsers when they visit a seemingly ordinary Web page. If the user's Internet Explorer security preferences are set to the lowest levels, the code can install spyware programs on the user's hard drive and even set them so that they launch automatically the next time the user reboots. It can also insert toolbars and other objects into the browser itself, essentially changing the way the browser works in the future-all without the user's permission.

❍ **When users click on a deceptive or confusing pop-up**: Some pop-up screens don't actually deliver advertisements but attempt to install unwanted software on your system and change your system configurations. These pop-ups can be very clever. Instead of "To install this program, click Yes," the prompt unexpectedly reads, "To install this program, click No." After clicking on these pop-ups, the user may find that the computer now displays new bookmarks and a different home page as well as having unwanted software installed.

❍ **During a peer-to-peer (P2P) file transfer or software download**: Some spyware hides out in group directories on P2P networks, such as music sharing networks, and then spreads by infecting machines as users search for music selections. Other spyware is bundled with software that the user is intentionally downloading or purchasing. Some of these programs are bundled so tightly that, once installed, they are nearly impossible to get rid of.

What Do Spyware And Adware Do?

Employees may not even know that their computers have been infected until they find ads popping up all over their desktops. Or one day they may notice that their computers are working slower than usual, which happens when spyware programs are uploading information to a remote server or are downloading new ads. These are only symp-

toms of what can be a very serious problem for an organisation.

Because spyware and adware exist as independent executable programs, these programs can monitor keystrokes, scan files on the hard drive, install other spyware programs, read cookies, and change the default home page on the Web browser. The programs continually relay this information back to the spyware author, who either uses it for advertising or marketing purposes or sells the information to another party.

Organisations whose very existence depends on protecting their valuable intellectual property cannot risk losing this competitive edge to information thieves. And organisations whose investors and clients rely on them to safeguard and protect personal, medical, and financial information, to name just a few, cannot afford to question whether critical information is being accessed by spyware. Organisations that need to demonstrate compliance with government regulations for information security are especially affected by spyware.

When spyware is part of the corporate computing environment, capturing confidential information or secretly perusing files and applications, regulatory compliance is virtually impossible. Even in computing environments that encrypt data, spyware remains a threat to the security of corporate data because its keystroke-logging components capture input before it can be encrypted.

Spyware and adware significantly increases the burden of IT Help Desk staff by causing application conflicts, malfunction of legitimate applications, and system instability. Many times, the IT Help Desk staff may have to re-image the desktops/laptops to completely get rid of problems caused by spyware.

When spyware and adware programs send information back to their home servers, they must connect to the Internet. In doing this, spyware can cause unexpected lockups and many other problems in Windows. When these events occur, calls to IT Help Desks increase as employees struggle to understand why their computers

are crashing or business applications are running more slowly.

Some spyware binds itself to key operating system files and modifies critical registry entries. Attempts to delete these files can limit or even disable the system's Internet connection capabilities. For example, WebHancer is a spyware program that automatically launches at Windows startup. It monitors Web sites being viewed and sends performance data back to WebHancer's servers. WebHancer has had conflicts with Microsoft IIS, causing problems with ASP scripts. It causes server script ASP pages to stop functioning when the Web application settings are in medium and high isolation modes. WebHancer has modified the computer's Windows Sockets configuration, binding itself to Winsock so that all packets are passed through WebHancer. Deleting WebHancer files may result in loss of ability to connect to the Internet. Employees who consider themselves sophisticated computer users may try to locate and delete spyware programs themselves, inadvertently creating even greater problems, such as the WebHancer problem described above.

Since spyware and adware are piggyback programs that run separately from the program they accompany, they use additional processing power, hard drive space, and network bandwidth. Spyware uses computer memory resources and consumes bandwidth as it sends information back to the spyware's home base via the user's Internet connection. Because spyware uses memory and system resources, the applications running in the background can lead to system crashes or general system instability. These files also consume a great deal of bandwidth and can create bottlenecks for critical business applications.

Having to close pop-up advertising windows and reset home pages that have been redirected by spyware is annoying and time consuming. Employee productivity is also affected by slow network performance and system instability. Many times, employees-unaware of the cause of their computer problems-contact the Help Desk frequently for support. This can seriously affect employee productivity and places an increased burden on Help Desk staff.

Businesses that wish to guard against spyware, adware and other unwanted applications will benefit from supplementing traditional protection methods (including firewalls, intrusion detection systems and antivirus programs) with new strategies that address the unique characteristics of spyware. A comprehensive, company-wide spyware-prevention strategy should include multiple elements:

1. Detailed Acceptable Use Policies (AUP) For Company-owned Computers:

An effective company-wide policy should specifically address the ways in which spyware may enter, including browsing to non-work-related sites, opening unsolicited e-mail attachments, and installing unauthorised and/or non-work-related applications. If such activities are allowed, the AUP should establish configuration and usage procedures that would help to protect the company against inadvertent pest installation. While AUPs are an effective employee education method, they are not sufficient on their own to protect against intentional or accidental violations. It's not sufficient to allow individual users to employ their own favourite anti-spyware products. Spyware can migrate from one PC to another attached to internal e-mails and other communications. The best approach is to use a networked anti-spyware solution that provides for a level of centralised management that ensure all your PCs and servers are covered and alerts the IT manager of spyware incursions.

2. Threat-specific Protection

Unauthorised third parties will always find new ways to access forbidden data or resources. As noted previously, anti-virus programs do not provide reliable protection against spyware; therefore, dedicated tools are required.

Businesses considering the deployment of company-wide protection against spyware, adware and other malicious applications will benefit from solutions that:

○ **Address legal and regulatory issues**: As a starting point, an effective strategy should address the local, national and international legislation regarding confidentiality and integrity of customer, financial and employee data.

○ **Minimise strain on computing resources**: The software should provide comprehensive protection against a variety of threats without consuming significant bandwidth or operating resources on servers and client computers.

○ **Decrease end user interaction**: Pest-prevention software should operate transparently so that employees cannot bypass or disable the protection.

○ **Reduce IT overhead**: To free IT staff to focus on more strategic projects, the software should offer automatic deployment, trickledown updates and centralised reporting and management.

○ **Enable flexible file handling**: Because there may be legitimate business uses for potentially suspect applications, such as file-sharing programs and network packet sniffers, administrators should be able to make case-by-case decisions about which kinds of tools may be allowed in specific circumstances.

○ **Support the improvement of company-wide protection**: The protection software should offer comprehensive event-logging capabilities so that administrators can spot trends and update acceptable use policies and firewall configurations accordingly.

II. Network Security-Related Issues

In the rapidly evolving world of network security, there's a thin line between paranoia and prudent protection. Hackers are growing in both number and sophistication, and the stakes are rising every day. New technologies have triggered a shift in the network security paradigm, expanding vulnerability exponentially.

Distributed personal firewalls are needed to protect corporate networks from Internet-enabled espionage, sabotage, and vandalism. Each individual PC—local and remote—must employ security technology to prevent known and unknown attacks. Real-world security needs to be flexible and make intelligent use of Policy Lifecycle Management to balance protection with productivity.

Enterprise Networks At Risk

As networks become larger, more complex, and more distributed, corporations face a growing vulnerability to hacker attacks and industrial espionage. Security consciousness and security spending are both on the increase, but not at a sufficient pace to stay ahead of the growing threat. The DefCon Internet Security site estimated that in 2002, approximately 19 million people had the skills to mount a cyber attack. According to a CSI/FBI survey, a new generation of profit-motivated hackers raised the stakes for corporate security managers. They used Trojan horses such as Back Orifice, Sub7, and other custom spyware to control remote machines, steal passwords, and compromise corporate networks. Hackers randomly scan for vulnerabilities and deploy viruses to harvest IP addresses and information.

Once inside, hackers can conduct espionage or sabotage, steal financial information, disrupt business, and cause public embarrassment. Even networks with VPN tunnels are at risk. The VPN will secure the data in transit, but leaves the endpoints vulnera-

ble. Data delivered safely can be harvested by Trojans at the exposed endpoints. Whether a hacker's goal is vandalism or illicit profit, the costs can be enormous. Computer Economics, an independent research firm, estimated global financial damage from malicious code in 2000 at $17.1 billion. mi2g, a London-based e-commerce research and development company, put the mark even higher, at $20 billion.

New technologies have triggered a paradigm shift in network security. In the old network model, almost all PCs connected to the Internet via a central gateway. Guarding the gateway effectively created a defensive perimeter. This model is no longer adequate.

First, while corporate networks shored up their security with centralised firewalls, anti-virus and intrusion detection, hackers exposed other vulnerabilities. Second, the explosion of remote and mobile users with always-on, broadband Internet connections to the network means most networks now have hundreds, or even thousands, of vulnerable 'backdoors'.

The Gartner Group noted, "Broadband connections are rife with threats to remote devices. Viruses, Trojan horses, zombies, keystroke monitoring, file shares and denial-of-service attacks all threaten the remote machine and, by extension, put the enterprise's IT resources at risk." Microsoft and others were hacked in this way. Incursions of this sort can quickly turn into high-profile PR disasters, or worse, go undetected for months before being exposed. Vulnerability has expanded exponentially. As companies go from single gateways to thousands of Internet connected endpoints, the number of vulnerabilities for networks has exploded. IDC reported the number of remote users in the year 2000 at 39 million and growing nine percent annually.

This accelerating trend is creating even more back doors. In addition, laptop users physically bypass the firewall every day, and wireless networks have no definable boundaries. Effectively, the network perimeter has disappeared. Hackers have taken notice, and so have government regulators: recent legislation requiring tighter security

in the healthcare and financial services industries is a telling sign of the times, and a reminder of how much we all have to lose.

Centrally-Managed Endpoint Security

Prolific threats require a pervasive solution. To reclaim peace of mind and control of the network perimeter, each endpoint must be secured. Distributed endpoint security, centrally managed personal firewalls, and application control technology offer the best defence against attacks that threaten corporate productivity, data, and reputation. IDC notes, ". . .as 'always-on' Internet access grows (with digital subscriber line [DSL] and cable modems) and as more companies allow telecommuting, the need for distributed and personal firewalls will grow." Consequently, Peter Lindstrom of the Hurwitz Group stated, "The personal firewall may well become more significant in the long run than the corporate firewall." Personal firewalls and application control can also help secure endpoints behind the corporate firewall, by preventing internal hacking, unknown Trojans, and spyware from exposing sensitive data outside the corporation.

Similar to a corporate network, each individual PC—local and remote—must employ multiple approaches to security technology. Only a policy-based, application-oriented distributed firewall, on each and every enterprise PC, can provide the protection needed to stop thousands of new and unknown hacking combinations and techniques.

For true endpoint security, a distributed firewall must incorporate the following functions:
- Obscure PCs to prevent outside access from hackers
- Prevent applications from becoming hacker tools by allowing only authenticated and approved applications to access the Internet
- Secure e-mail attachments to prevent e-mail from being used as a transmission tool for viruses and malicious worms
- Block, alert and log intrusions
- Provide cooperative gateway protection to leverage the existing IS infrastructure and ensure that only endpoints with distributed firewalls and current security policy access the network

Combined, these security functions and others protect each indi
vidual PC—local and remote. By distributing and enforcing PC
security and security policy across all endpoints, the chances of
a security breach are greatly reduced, thereby offering greater pro
tection to the entire network. Real-world security has to be flexi
ble: threats, organisations, and corporate networks change. It is a
fact of life that attackers will learn and adapt in an attempt to cir
cumvent defences. And, policy that once supported productivity
may later thwart it. Real-world security solutions have to evolve to
respond to new threats and changing organisational needs.

Flexible security policy management is critical for maintaining
maximum corporate security. Policy Lifecycle Management is the
key to maximizing corporate security and productivity. Centrally
managed policy provides an enforcement mechanism to ensure all
endpoints are compliant. Policy Lifecycle Management optimises
security by streamlining policy creation and providing feedback,
enforcing and updating policy at all times.

III. Network Security: Best Practices

Without a security policy, the availability of your network can be compromised. The policy begins with assessing the risk to the network and building a team to respond. Continuation of the policy requires implementing a security change management practice and monitoring the network for security violations. Last, the review process modifies the existing policy and adapts to lessons learned.

We look here at preparation, prevention, and response, in detail.

Preparation

Prior to implementing a security policy, you must:
❍ Create usage policy statements
❍ Conduct a risk analysis
❍ Establish a security team structure

1. Creating Usage Policy Statements

Creating usage policy statements that outline users' roles and responsibilities with regard to security is recommended. You can start with a general policy that covers all network systems and data within your company. This chapter should provide the general user community with an understanding of the security policy, its purpose, guidelines for improving their security practices, and definitions of their security responsibilities. If your company has identified specific actions that could result in punitive or disciplinary actions against an employee, these actions and how to avoid them should be clearly articulated here.

The next step is to create a partner-acceptable use statement to provide partners with an understanding of the information that is available to them, the expected disposition of that information, as well as the conduct of the employees of your company. You should clearly explain any specific acts that have been identified as secu-

rity attacks and the punitive actions that will be taken should a security attack be detected.

Last, create an administrator-acceptable use statement to explain the procedures for user account administration, policy enforcement, and privilege review. If your company has specific policies concerning user passwords or subsequent handling of data, clearly present those policies as well. Check the policy against the partner-acceptable use and the user acceptable use policy statements to ensure uniformity. Make sure that administrator requirements listed in the acceptable use policy are reflected in training plans and performance evaluations.

2. Conduct A Risk Analysis

A risk analysis should identify the risks to your network, network resources, and data. This doesn't mean you should identify every possible entry point to the network, nor every possible means of attack. The intent of a risk analysis is to identify portions of your network, assign a threat rating to each portion, and apply an appropriate level of security. This helps maintain a workable balance between security and required network access. Assign each network resource one of the following three risk levels:

❍ Low Risk Systems or data that if compromised (data viewed by unauthorised personnel, data corrupted, or data lost) would not disrupt the business or cause legal or financial ramifications. The targeted system or data can be easily restored and does not permit further access of other systems.

❍ Medium Risk Systems or data that if compromised (data viewed by unauthorised personnel, data corrupted, or data lost) would cause a moderate disruption in the business, minor legal or financial ramifications, or provide further access to other systems. The targeted system or data requires a moderate effort to restore or the restoration process is disruptive to the system.

❍ High Risk Systems or data that if compromised (data viewed by unauthorised personnel, data corrupted, or data lost) would cause an extreme disruption in the business, cause major legal

or financial ramifications, or threaten the health and safety of a person. The targeted system or data requires significant effort to restore or the restoration process is disruptive to the business or other systems.

Assign a risk level to each of the following: core network devices, distribution network devices, access network devices, network monitoring devices, network security devices, e-mail systems, network file servers, network print servers, network application servers (DNS and DHCP), data application servers (Oracle or other standalone applications), desktop computers, and other devices (standalone print servers and network fax machines). Network equipment such as switches, routers, DNS servers, and DHCP servers can allow further access into the network, and are therefore either medium or high risk devices. It is also possible that corruption of this equipment could cause the network itself to collapse. Such a failure can be extremely disruptive to the business.

Once you've assigned a risk level, it's necessary to identify the types of users of that system. The five most common types of users are:

❍ Administrators Internal users responsible for network resources
❍ Privileged Internal users with a need for greater access
❍ Users Internal users with general access
❍ Partners External users with a need to access some resources
❍ Others External users or customers

The identification of the risk level and the type of access required of each network system forms the basis of the following security matrix. The security matrix provides a quick reference for each system and a starting point for further security measures, such as creating an appropriate strategy for restricting access to network resources.

3. Establish A Security Team Structure

Create a cross-functional security team led by a Security Manager with participants from each of your company's operational areas. The representatives on the team should be aware of the security policy and the technical aspects of security design and implemen-

System	Description	Risk Level	Types of Users
ATM switches	Core network device	High	Administrators for device configuration (support staff only); All others for use as a transport
Network Routers	Distribution Network Device	High	Administrators for device configuration (support staff only); All others for use as a transport
Closet Switches	Access Network Device	Medium	Administrators for device configuration (support staff only); All others for use as a transport
ISDN or dial-up servers	Access Network Device	Medium	Administrators for device configuration (support staff only); Partners and privileged users for special access
Firewall	Access Network Device	High	Administrators for device configuration (support staff only); All others for use as a transport.
DNS and DHCP servers	Network Applications	Medium	Administrators for configuration; General and privileged users for use
External e-mail Server	Network Application	Low	Administrators for configuration; All others for mail transport between the Internet and the internal mail server
Internal E-mail Server	Network Application	Medium	Administrators for configuration; All other internal users for use
Oracle Database	Network Application	Medium or High	Administrators for system administration; Privileged users for data updates; General users for data access; All others for partial data access

tation. Often, this requires additional training for the team members. The security team has three areas of responsibilities: policy development, practice, and response. Policy development is focused on establishing and reviewing security policies for the company. At a minimum, review both the risk analysis and the security policy on an annual basis. Practice is the stage during which the security team conducts the risk analysis, the approval of security change requests, reviews security alerts from both vendors and the CERT mailing list, and turns plain language security policy requirements into specific technical implementations.

The last area of responsibility is response. While network monitoring often identifies a security violation, it is the security team members who do the actual troubleshooting and fixing of such a violation. Each security team member should know in detail the security features provided by the equipment in his or her operational area.

While we have defined the responsibilities of the team as a whole, you should define the individual roles and responsibilities of the security team members in your security policy.

Prevention

Prevention can be broken into two parts: approving security changes and monitoring security of your network.

1. Approving Security Changes

Security changes are defined as changes to network equipment that have a possible impact on the overall security of the network. Your security policy should identify specific security configuration requirements in non-technical terms. In other words, instead of defining a requirement as "No outside sources FTP connections will be permitted through the firewall", define the requirement as "Outside connections should not be able to retrieve files from the inside network". You'll need to define a unique set of requirements for your organisation.

The security team should review the list of plain language requirements to identify specific network configuration or design

issues that meet the requirements. Once the team has created the required network configuration changes to implement the security policy, you can apply these to any future configuration changes. While it's possible for the security team to review all changes, this process allows them to only review changes that pose enough risk to warrant special treatment.

We recommend that the security team review the following types of changes:

❍ Any change to the firewall configuration
❍ Any change to access control lists (ACL)
❍ Any change to Simple Network Management Protocol (SNMP) configuration
❍ Any change or update in software that differs from the approved software revision level list

It's also recommended to adhere to the following guidelines:

❍ Change passwords to network devices on a routine basis
❍ Restrict access to network devices to an approved list of personnel
❍ Ensure that the current software revision levels of network equipment and server environments are in compliance with the security configuration requirements

In addition to these approval guidelines, have a representative from the security team sit on the change management approval board, in order to monitor all changes that the board reviews. The security team representative can deny any change that is considered a security change until it has been approved by the security team.

2. Monitoring Your Network Security

Security monitoring is similar to network monitoring, except that it focuses on detecting changes in the network that indicate a security violation. The starting point for security monitoring is determining what is a violation. In 'Conduct a Risk Analysis', we identified the level of monitoring required based on the threat to the system. In 'Approving Security Changes', we identified specific threats to the network. By looking at both these parameters, we'll develop a clear picture of what you need to monitor and how often.

In the Risk Analysis matrix, the firewall is considered a high-risk network device, which indicates that you should monitor it in real time. From the Approving Security Changes section, you see that you should monitor for any changes to the firewall. This means that the SNMP polling agent should monitor such things as failed login attempts, unusual traffic, changes to the firewall, access granted to the firewall, and connections setup through the firewall.

Following this example, create a monitoring policy for each area identified in your risk analysis. It's recommended to monitor low-risk equipment weekly, medium-risk equipment daily, and high-risk equipment hourly. If you require more rapid detection, monitor on a shorter time frame. Last, your security policy should address how to notify the security team of security violations. Often, your network monitoring software will be the first to detect the violation. It should trigger a notification to the operations centre, which in turn should notify the security team, using a pager if necessary.

3. Response

Response can be broken into three parts: security violations, restoration, and review.

Security Violations: When a violation is detected, the ability to protect network equipment, determine the extent of the intrusion, and recover normal operations depends on quick decisions. Having these decisions made ahead of time makes responding to an intrusion much more manageable.

The first action following the detection of an intrusion is the notification of the security team. Without a procedure in place, there will be considerable delay in getting the correct people to apply the correct response. Define a procedure in your security policy that is available 24 hours a day, 7 days a week. Next you should define the level of authority given to the security team to make changes, and in what order the changes should be made.

Possible corrective actions are:

❍ Implementing changes to prevent further access to the violation
❍ Isolating the violated systems
❍ Contacting the carrier or ISP in an attempt to trace the attack
❍ Using recording devices to gather evidence
❍ Disconnecting violated systems or the source of the violation
❍ Contacting the police, or other government agencies

❍ Shutting down violated systems
❍ Restoring systems according to a prioritised list
❍ Notifying internal managerial and legal personnel
  Be sure to detail any changes that can be conducted without management approval in the security policy. Last, there are two reasons for collecting and maintaining information during a security attack: to determine the extent to which systems have been compromised by a security attack, and to prosecute external violations. The type of information and the manner in which you collect it differs according to your goal. To determine the extent of the violation, do the following:

❍ Record the event by obtaining sniffer traces of the network, copies of log files, active user accounts, and network connections
❍ Limit further compromise by disabling accounts, disconnecting network equipment from the network, and disconnecting from the Internet
❍ Backup the compromised system to aid in a detailed analysis of the damage and method of attack
❍ Look for other signs of compromise. Often when a system is compromised, there are other systems or accounts involved
❍ Maintain and review security device log files and network monitoring log files, as they often provide clues to the method of attack

  If you're interested in taking legal action, have your legal department review the procedures for gathering evidence and involvement of the authorities. Such a review increases the effectiveness of the evidence in legal proceedings. If the violation was internal in nature, contact your Human Resources department.

Restoration: Restoration of normal network operations is the final goal of any security violation response. Define in the security policy how you conduct, secure, and make available normal backups. As each system has its own means and procedures for backing up, the security policy should act as a meta-policy, detailing for each system the security conditions that require restoration from backup. If approval is required before restoration can be done, include the process for obtaining approval as well.

Review: The review process is the final effort in creating and maintaining a security policy. There are three things you'll need to review: policy, posture, and practice. The security policy should be a living document that adapts to an ever-changing environment. Reviewing the existing policy against known Best Practices keeps the network up to date. Also, check the CERT Web site (http://www.cert.org) for useful tips, practices, security improvements, and alerts that can be incorporated into your security policy.

You should also review the network's posture in comparison with the desired security posture. An outside firm that specialises in security can attempt to penetrate the network and test not only the posture of the network, but the security response of your organisation as well. For high-availability networks, it's recommended to conduct such a test annually.

Finally, practice is defined as a drill or test of the support staff to insure that they have a clear understanding of what to do during a security violation. Often, this drill is unannounced by management and done in conjunction with the network posture test. This review identifies gaps in procedures and training of personnel so that corrective action can be taken.

IV. The Need For Secure Operating Systems

Source: "The Inevitability of Failure: The Flawed Assumption of Security in Modern Computing Environments"
available at
http://jya.com/paperF1.htm

Public awareness of the need for security in computing systems is growing as critical services are becoming increasingly dependent on interconnected computing systems. National infrastructure components such as the electric power, telecommunication and transportation systems can no longer function without networks of computers. The advent of the World Wide Web has especially increased public concern for security. Security is the primary concern of businesses that want to use the Internet for commerce and maintaining business relationships.

The increased awareness of the need for security has resulted in an increase of efforts to add security to computing environments. However, these efforts suffer from the flawed assumption that security can adequately be provided in application space without certain security features in the operating system. In reality, operating system security mechanisms play a critical role in supporting security at higher levels. Yet today, debate persists in the research community as to what role operating systems should play in secure systems. The computer industry has not accepted the critical role of the operating system to security, as evidenced by the inadequacies of the basic protection mechanisms provided by current mainstream operating systems.

The necessity of operating system security to overall system security is undeniable; the underlying operating system is responsible for protecting application-space mechanisms against tampering, bypassing, and spoofing attacks. If it fails to meet this responsibility, system-wide vulnerabilities will result.

The need for secure operating systems is especially crucial in today's computing environment. Substantial increases in connectivity and data sharing have increased the risk to systems such that even a careful and knowledgeable user running on a single-user system is no longer safe from the threat of malicious code. Because the distinction between data and code is vanishing, malicious code may be introduced, without a conscious decision on the part of a user to install executable code, whenever data is imported into the system. For example, malicious code

could be introduced with a Java applet or by viewing apparently benign data that, in actuality, contains executable code. More so than ever, secure operating systems are needed to protect against this threat.

Here, we identify some features of secure operating systems that are necessary to protect application-space security mechanisms yet are lacking in mainstream operating systems. They form the 'missing link' of security. Although this section only deals with features, it is important to note that features alone are inadequate. Assurance evidence must be provided to demonstrate that the features meet the desired system security properties and to demonstrate that the features are implemented correctly. Assurance is the ultimate missing link; although approaches to providing assurance may be controversial, the importance of assurance is undeniable.

Mandatory Security

An operating system's mandatory security policy may be divided into several kinds of policies, such as an access control policy, an authentication usage policy, and a cryptographic usage policy. A mandatory access control policy specifies how subjects may access objects under the control of the operating system. A mandatory authentication usage policy specifies what authentication mechanisms must be used to authenticate a principal to the system. A mandatory cryptographic usage policy specifies what cryptographic mechanisms must be used to protect data. Additionally, various subsystems of the operating system may have their own mechanism usage policies. These subsystem specific usage policies may be dependent on the cryptographic usage policy. For example, a network usage policy for a router might specify that sensitive network traffic should be protected using IPSEC ESP in tunnelling mode prior to being sent to an external network. The selection of a cryptographic algorithm for IPSEC ESP may be deferred to the cryptographic usage policy.

A secure system must provide a framework for defining the operating system's mandatory security policy and translating it to

a form interpretable by the underlying mandatory security mechanisms of the operating system. Without such a framework, there can be no real confidence that the mandatory security mechanisms will provide the desired security properties. An operating system that provides mandatory security may nonetheless suffer from the presence of high-bandwidth covert channels. This is an issue whenever the mandatory security policy is concerned with confidentiality. This should not, however, be a reason to ignore mandatory security. Even with covert channels, an operating system with basic mandatory controls improves security by increasing the required sophistication of the adversary.

Once systems with basic mandatory controls become mainstream, covert channel exploitation will become more common and public awareness of the need to address covert channels in computing systems will increase.

In any system that supports mandatory security, some applications require special privileges in the mandatory policy in order to perform some security-relevant function. Such applications are frequently called trusted applications because they are trusted to correctly perform some security-related function and because they are trusted to not misuse privileges required in order to perform that function. If the mandatory security mechanisms of a secure operating system only support coarse-grained privileges, then the security of the overall system may devolve to the security of the trusted applications on the system. To reduce the dependency on trusted applications, the mandatory security mechanisms of an operating system should be designed to support the principle of least privilege.

Type enforcement is an example of a mandatory security mechanism that may be used both to limit trusted applications to the minimal set of privileges required for their function and to confine the damage caused by any misuse of these privileges.

The mandatory security mechanisms of an operating system may be used to support security-related functionality in applica-

tions by rigorously ensuring that subsystems are unbypassable and tamperproof. For example, type enforcement may be used to implement assured pipelines to provide these properties. An assured pipeline ensures that data flowing from a designated source to a designated destination must pass through a security-related subsystem and ensures the integrity of the subsystem. Many of the security requirements of these applications may be ensured by the underlying mandatory security mechanisms of the operating system.

Operating system mandatory security mechanisms may also be used to rigorously confine an application to a unique security domain that is strongly separated from other domains in the system. Applications may still misbehave, but the resulting damage can now be restricted to within a single security domain. This confinement property is critical to controlling data flows in support of a system security policy. In addition to supporting the safe execution of untrustworthy software, confinement may support functional requirements, such as an isolated testing environment in an insulated development environment.

Although one could attempt to enforce a mandatory security policy through discretionary security mechanisms, such mechanisms can not defend against careless or malicious users. Since discretionary security mechanisms place the burden for security on the individual users, carelessness by any one user at any point in time may lead to a violation of the mandatory policy. In contrast, mandatory security mechanisms limit the burden to the system security policy administrator. With only discretionary mechanisms, a malicious user with access to sensitive data and applications may directly release sensitive information in violation of the mandatory policy. Although that same user may also be able to leak sensitive information in ways that do not involve the computing system, the ability to leak the information through the computing system may increase the bandwidth of the leak and may decrease its traceability. In contrast, with mandatory security mechanisms, he may only leak sensitive information through covert channels, which limits the bandwidth and increases accountability, if covert channels are audited.

Furthermore, even with users who are benign and careful, the mandatory security policy may still be subverted by flawed or malicious applications when only discretionary mechanisms are used to enforce it. The distinction between flawed and malicious software is not particularly important in this paper. In either case, an application may fail to apply security mechanisms required by the mandatory policy or may use security mechanisms in a way that is inconsistent with the user's intent. Mandatory security mechanisms may be used to ensure that security mechanisms are applied as required and can protect the user against inadvertent execution of untrustworthy applications.

Although the user may have carefully defined the discretionary policy to properly implement the mandatory policy, an application may change the discretionary policy without the user's approval or knowledge. In contrast, the mandatory policy may only be changed by the system security policy administrator.

In the case of personal computing systems, where the user may be the system security policy administrator, mandatory security mechanisms are still helpful in protecting against flawed or malicious software. In the simplest case, where there is only a distinction between the user's ordinary role and the user's role as system security policy administrator, the mandatory security mechanisms can protect the user against unintentional execution of untrustworthy software. With a further subdivision of the user's ordinary role into various roles based on function, mandatory security mechanisms can confine the damage that may be caused by flawed or malicious software.

Although there are a number of commercial operating systems with support for mandatory security, none of these systems have become mainstream. These systems have suffered from a fixed notion of mandatory security, thereby limiting their market appeal. Furthermore, these systems typically lack adequate support for constraining trusted applications. In order to reach a wider market, operating systems must support a more general

notion of mandatory security and must support flexible configuration of mandatory policies.

Mainstream commercial operating systems rarely support the principle of least privilege even in their discretionary access control architecture. Many operating systems only provide a distinction between a completely privileged security domain and a completely unprivileged security domain. Even in Microsoft Windows NT, the privilege mechanism fails to adequately protect against malicious programs because it does not limit the privileges that a program inherits from the invoking process based on the trustworthiness of the program.

Current microkernel-based research operating systems have tended to focus on providing primitive protection mechanisms which may be used to flexibly construct a higher-level security architecture. Many of these systems use kernel-managed capabilities as the underlying protection mechanism. However, typical capability architectures are inadequate for supporting mandatory access controls with a high degree of flexibility and assurance. Flask, a variant of the Fluke microkernel, provides a mandatory security framework similar to that of DTOS, a variant of the Mach microkernel; both systems provide mechanisms for mandatory access control and a mandatory policy framework.

Trusted Paths

A trusted path is a mechanism by which a user may directly interact with trusted software, which can only be activated by either the user or the trusted software and may not be imitated by other software. In the absence of a trusted path mechanism, malicious software may impersonate trusted software to the user or may impersonate the user to trusted software. Such malicious software could potentially obtain sensitive information, perform functions on behalf of the user in violation of the user's intent, or trick the user into believing that a function has been invoked without actually invoking it. In addition to supporting trusted software in the base system, the trusted path mechanism should be extensible to support the subsequent addition of trusted applications by a sys-

tem security policy administrator.

The concept of a trusted path can be generalised to include inter-actions beyond just those between trusted software and users. The TNI introduces the concept of a trusted channel for communication between trusted software on different network components. More generally, a mechanism that guarantees a mutually authenticated channel, or protected path, is necessary to ensure that critical system functions are not being spoofed. Although a protected path mecha-nism for local communications could be constructed in application space without direct authentication support in the operating sys-tem, it is preferable for an operating system to provide its own pro-tected path mechanism since such a mechanism will be simpler to assure and is likely to be more efficient.

Most mainstream commercial operating systems are utterly lacking in their support for either a trusted path mechanism or a protected path mechanism. Microsoft Windows NT does provide a trusted path for a small set of functions such as login authentica-tion and password changing but lacks support for extending the trusted path mechanism to other trusted applications. For local communications, NT does provide servers with the identity of their clients; however, it does not provide the server identity to the client.

General Examples

Without operating system support for mandatory security and trusted path, application space mechanisms for access control and cryptography cannot be implemented securely.

Access Control: An application-space access control mechanism may be decomposed into an enforcer component and a decider component. When a subject attempts to access an object protected by the mechanism, the enforcer component must invoke the decider component, supplying it with the proper input parame-ters for the policy decision, and must enforce the returned deci-sion. A common example of the required input parameters is the security attributes of the subject and the object. The decider com-ponent may also consult other external sources in order to make the policy decision. For example, it may use an external policy

database and system information such as the current time. If a malicious agent can tamper with any of the components in the access control mechanism or with any inputs to the decision, then the malicious agent can subvert the access control mechanism. Even if the components and all of the inputs are collocated within a single file, the operating system security mechanisms are still relied upon to protect the integrity of that file. As discussed, only mandatory security mechanisms can rigorously provide such integrity guarantees.

Even with strong integrity guarantees for the policy decision inputs, if an authorised user invokes malicious software, the malicious software could change an object's security attributes or the policy database's rules without the user's knowledge or consent. The access control mechanism requires a trusted path mechanism in the operating system in order to ensure that arbitrary propagation of access cannot occur without explicit authorisation by a user.

If a malicious agent can impersonate the decider component to the enforcer component, or if a malicious agent can impersonate any source of inputs to the decision, then the malicious agent can subvert the mechanism. If any of the components or external decision input sources are not collocated within a single application, then the access control mechanism requires a protected path mechanism. If a malicious agent can bypass the enforcer component, then it may trivially subvert the access control mechanism. Mandatory security mechanisms in the operating system may be used to ensure that all accesses to the protected objects are mediated by the enforcer component.

Cryptography: An analysis of application-space cryptography may be decomposed into an analysis of the invocation of the cryptographic mechanism and an analysis of the cryptographic mechanism itself. As an initial basis for discussion, suppose that the cryptographic mechanism is a hardware token that implements the necessary cryptographic functions correctly and that there is a secure means by which the cryptographic keys are established in

the token. Even in this simplified case, where the confidentiality and integrity of algorithms and keys is achieved without operating system support, we will demonstrate that there are still vulnerabilities which may only be effectively addressed with the features of a secure operating system.

One vulnerability in this simplified case is that invocation of the token cannot be guaranteed. Any legitimate attempt to use the token might not result in a call to the token. The application that performs the cryptographic invocation might be bypassed or modified by malicious applications or malicious users. Malicious applications might impersonate the cryptographic token to the invoking application. Mandatory security and protected path features in the operating system address this vulnerability. Mandatory security mechanisms may be used to ensure that the application that invokes the cryptographic token is unbypassable and tamperproof against both malicious software and malicious users. Unbypassability could also be achieved by using an inline cryptographic token, which is physically interposed between the sender of the data to be protected and the receiver of the protected data; however, this would be less flexible. A protected path mechanism may be used to ensure that malicious software cannot impersonate the cryptographic token to the invoking application. Misuse of the cryptographic token is a second vulnerability in the simplified case. Misuse may involve the use of a service, algorithm, session or key by an unauthorised application. Without operating system support for identifying callers, a cryptographic token can do little more than require that a user activate it, after which, any service, algorithm, session or key authorised for that user may be used by any application on the system. In this case, the cryptographic token may be misused by applications operating on behalf of other users or may be misused by malicious software operating on behalf of the authorised user.

Furthermore, unless the cryptographic token has a direct physical interface for user activation, malicious software can spoof the token to the user, obtain authentication information, and subsequently activate the cryptographic token without the user's

knowledge or consent. Even with a direct physical interface to the user, it is impractical for the cryptographic token to require user confirmation for every cryptographic operation.

This second vulnerability may be addressed through mandatory security, trusted path and protected path features in the operating system. A trusted path mechanism obviates the need for a separate physical interface for activation. A protected path mechanism permits the cryptographic token to identify its callers and enforce fine-grained controls over the use of services, algorithms, sessions and keys. As an alternative to having the token deal with fine-grained controls over its usage, mandatory security mechanisms may also be used to provide such controls. For example, mandatory security mechanisms may be used to isolate the token for use only by applications executed by the user who activated the token. Furthermore, the mandatory security mechanisms can reduce the risk of malicious software being able to use the cryptographic token and may consequently limit the use of the trusted path mechanism to highly sensitive actions.

Hence, even in the simplest case, the features of a secure operating system are crucial to addressing the vulnerabilities of application-space cryptography. In the remainder of this section, the assumptions of the simplified case are removed, and the additional vulnerabilities are examined.

If the assumption that initial keys are securely established within the token is removed, then there is the additional vulnerability that the initial keys may be observed or modified by an unauthorised entity. Unless the initial keys are provided via a dedicated physical interface to the cryptographic token, the operating system must protect the path between the initial key source and the cryptographic token and may need to protect the initial key source itself. Mandatory security mechanisms may be used to rigorously protect the path and the key source. A trusted path may be required for initial keying.

If the assumption that the cryptographic mechanism is con-

fined to a single hardware token is removed and implemented in software instead, the confidentiality and integrity of the cryptographic mechanism's code and data becomes dependent on the operating system, including both memory protection and file protection. Mandatory security is needed to rigorously ensure the mechanism's integrity and confidentiality. If any external inputs, such as input parameters to a random number generator, are used by the cryptographic mechanism, the input sources and the path between the input sources and the cryptographic mechanism must be protected with mandatory security mechanisms.

System Security: No single technical security solution can provide total system security; a proper balance of security mechanisms must be achieved. Each security mechanism provides specific security functions and should be designed to only provide those functions. It should rely on other mechanisms for support and for required security services. In a secure system, the entire set of mechanisms complement each other so that they collectively provide a complete security package. Systems that fail to achieve this balance will be vulnerable.

A secure operating system is an important and necessary piece to the total system security puzzle, but it is not the only piece. A highly secure operating system would be insufficient without application-specific security built upon it. Certain problems are actually better addressed by security implemented above the operating system. One such example is an electronic commerce system that requires a digital signature on each transaction. A application-space cryptographic mechanism in the transaction system protected by secure operating system features might offer the best system security solution.

No single security mechanism is likely to provide complete protection. Unsolved technical problems, implementation errors and flawed environmental assumptions will result in residual vulnerabilities. As an example, covert channels remain a serious technical challenge for secure operating system designers. These limitations must be understood, and suitable measures must be taken to

deploy complementary mechanisms designed to compensate for such problems. In the covert channel example, auditing and detection mechanisms should be utilised to minimise the chances that known channels are exploited. In turn, these should depend on secure operating systems to protect their critical components, such as audit logs and intrusion sensors, because they are subject to the same types of vulnerabilities as those discussed elsewhere here.

Virus And Malicious Code Protection For Wireless Devices

Source: "Virus and Malicious Code Protection for Wireless Devices"
available at
http://download.antivirus.com/ftp/white/wireless_protection022801.doc

Although malicious code has yet to cause serious damage or incur substantial costs in the wireless arena, such code seen in the lab and, in some cases, in the real world, has indicated that this undesirable code has the potential for serious disruption to the wireless infrastructure. As the line between cellular phones and personal digital assistants blurs, the enhanced functionality of the wireless devices that emerge offers a playground for hackers and e-vandals—in much the same way that each new medium emerging in the last two decades has offered such an opportunity.

The world is going mobile. While the lack of affordable mobile phone service is a fairly recent memory for many consumers, today, most consumers take for granted the ability to communicate with friends and family anywhere, anytime, at a reasonable cost. At the same time, mobility is the watchword today in business. Global prosperity and an even faster pace of business are driving the desire for employees, partners, and customers to be able to communicate, without regard for location.

Yet increasingly today, mobility has a different face. The ability to transmit and receive wireless data is enabling an entirely new type of business. M-commerce, perhaps initially visualised by many as the teenager purchasing a soda using a cell phone in a

recent television commercial, is becoming the new way to purchase goods and services, transfer funds, and perform other types of wireless transactions.

The migration from simple voice communication to data communication is underway in earnest. According to Cahners In-Stat Group2, the most successful wireless data system is the short message service (SMS) on Global System for Mobile Communications (GSM) networks. Cahners points out that in a single month early this year, users sent 8 billion SMS messages worldwide. Interestingly, for many users in some parts of the world (including most notably, Japan), the wireless device is the most prevalent mode of accessing the Internet, compared to PC Internet access. More than 200 million SMS subscribers already dot the globe, and Cahners projected 742 million worldwide wireless Internet subscribers in 2004 and 607 million SMS subscribers in the same year.

Overview of Threats and Potential Damage

Yet, like each new communication and computing medium before it, wireless voice and data communication presents the opportunity for less desirable applications. The rapid spread of wireless communications presents new opportunities for hackers, disgruntled employees, and others to prove their prowess in spreading viruses and malicious code.

On the surface, the vulnerability of wireless devices to viruses and malicious code threats appears to follow the same patterns of vulnerabilities that the wired world has experienced. Yet, upon closer inspection, the vulnerabilities are more numerous and complex. Such threats can be categorised into three groups:

❍ Application-based threats
❍ Content-based threats
❍ Mixed threats (a power-packed combination of application and content-based threats not yet seen in the real world)

Application-based Threats

In the wireless world, application-based threats are posed by exe-

cutable malicious code that latches on to existing, or new, wireless applications. Application-based threats are potentially present anytime a software program is downloaded to, or executed on, a wireless device particularly when the program is downloaded or received from an unknown source. In the wired world, these threats are roughly analogous to the early viruses borne by executable programs (which were later superseded by the rise in Macro viruses—malicious code borne by non-executable files).

The first malicious application-based program that specifically targeted the Palm operating system (OS) used in Palm Pilot personal digital assistants (PDAs) was called 'Liberty Crack'. The free software, which could be downloaded from a Web site or accessed via Internet relay chat (IRC) rooms, pretended to convert the shareware Liberty Game Boy program into a registered version. When the program was executed, the user was not aware that, in the background, the program was deleting all executable applications in the handheld device. Liberty Crack did not affect the underlying Palm operating system or the embedded applications.

Liberty Crack and similar 'Trojan horses' are likely to spread very slowly 'in the wild' (i.e., in the real world) and represent a relatively low threat. Liberty Crack is designated a Trojan horse as it masquerades with one purpose, while harbouring a surprise purpose (similar to the Trojan horse of ancient Greece in which soldiers hid inside a hollow wooden horse presented as a gift by the Trojans).

While actual incidences of Liberty Crack have not been encountered in the wild, this Trojan horse is significant in its proof of concept—demonstrating that malicious code can be downloaded and may adversely impact PDAs. Many analysts have labelled Liberty Crack, which first made news in late August 2000, as a harbinger of more malicious code to come. For example, future wireless Trojans could steal data such as address book information, portal passwords, and other confidential information.

An independent developer for Palm computers, know as "Ardiri," assumed credit for designing Liberty Crack, saying its orig-

inal purpose was to clean up redundant data files. After providing the program to a few friends, Ardiri witnessed its proliferation within the Palm developer community, which then numbered about 80,000. Seeing that he may have caused a problem, he posted warnings about Liberty Crack on various Palm developer sites.

This evolution and proliferation of the Trojan horse raises two key aspects of application-based threats. First, it illustrates the potential for proliferation of malicious code, especially in the form of a Trojan, when it is disguised as a program with perceived value that is offered for free. Second, this early case reminds us that operating systems in widest use are likely to be the initial playgrounds of writers of malicious code. The large number of shareware applications available and the growing number of legitimate code developers in the community increases the likelihood of malicious behaviour. Further, the large number of possible affected users raises the potential profile of any malicious activity—an enticement for those seeking the limelight for destructive activities.

Since the discovery of Liberty Crack, antivirus experts such as Trend Micro have been tracking a number of other application-based, potentially destructive Palm programs, including Palm Phage—the first known virus designed to affect Palm PDAs. First seen about one month after Liberty, Palm Phage infects all third-party application programs when executed. Instead of running normally, infected executable files infect other third-party applications programs. Palm Phage can theoretically spread to other machines when the Palm is synchronised with a PC or when a Palm beams data via an infrared link to another Palm.

At about the same time, several joke programs were observed on PDAs that operate on the EPOC operating system. Little more than nuisances, these programs (e.g., EPOC_Alone.A and EPOC_Ghost.A) disturb users by sounding an alarm or flashing lights on the EPOC-enabled device. While these programs do not spread from device to device, they demonstrate that malicious code can cause bothersome disturbances on wireless devices.

Furthermore, the wireless world is seeing the regular birth of new technologies, with more on the horizon. Some of these technologies will expand the functionality of the device while others will dramatically change their connectivity with other devices (e.g., Bluetooth technology).

No users have lost data as a result of Palm Phage and the EPOC joke programs. But this malicious code ups the ante for such code in the wireless arena—demonstrating that self-replicating viruses are not only possible to develop, but easy to develop. And with the expanded functionality of these devices in the coming months and years, so will expand the potential for new threats from malicious code.

Content-based Threats

In content-based threats, the content (e.g., derogatory messages) is the threat, or malicious use of the content is the threat (e.g., spamming of e-mail). While e-mail has become the 'killer app' of the wireless world, it is also one of the most vulnerable to attack. Hence, the most common content-based threats to the wireless infrastructure occur through infected e-mail or spam mail.

The first content-based Trojan to attack wireless devices occurred in June 2000 with the appearance, in the wild, of the Visual Basic Script (VBS) Timofonica on the wireless network of Madrid, Spain-based Telefonica SA. Timofonica spread by sending infected e-mail messages from affected computers. When an infected e-mail reached a PC, it used Microsoft Outlook 98 or 2000 to send a copy of itself via infected e-mails to all addresses in the MS Outlook Address Book. This enabled the Trojan to spread quite rapidly. In the wired world, this behaviour is similar to that of the "ILoveYou" e-mail virus that caused worldwide damage estimated as high as $700 million in May 2000.

But Timofonica was more than an e-mail virus. For each e-mail it sent, the Trojan also dispatched an SMS message to a randomly generated address at the "correo.movistar.net" Internet host (see Figure 4). Since this host sends SMS messages to mobile

phones operating on the European GSM standard (the phone number is the prefix of the e-mail address in the message), the Trojan tried to spam people with SMS messages—in this case a derogatory depiction of Spanish telecom provider Telefonica Moviles.

Like the Liberty Crack Trojan, the Timofonica attack was benign and caused little damage. Although the program reached out into the wireless world, it propagated via land-based PCs and e-mails, not from phone to phone directly. Nevertheless, Timofonica demonstrated in-the-wild, the ability of malicious code to tap into the wireless infrastructure and spread with great speed. Timofonica had the potential to flood the wireless network with messages, reducing its performance or even impairing its ability to meet load. Worse, for wireless users billed on a per-message basis, receiving spam costs them money. A similar program was observed on Japan's ambitious I-mode system. Japan's largest cellular phone maker, NTT DoCoMo, developed and owns the I-mode system which appears to have successfully captured both consumer and business markets for wireless device transactions, wireless Internet access, and instant messaging in Japan. With more than 10 million users only 18 months after its launch, some analysts see I-mode as a feasible alternative to WAP being used in Europe and touted in North America.

In June 2000, a piece of malicious code began to send a particular message to wireless users on the I-mode system. When the user received the message and clicked on a hypertext link, the program dialled 110—the Japanese equivalent of 911 in North America—without the prior knowledge of the user. This loading of emergency service lines with useless calls demonstrated the ability of malicious code to reach out to other key infrastructures and cause serious damage. Another potential content-based threat that may soon enter the wireless world, as wireless devices become more sophisticated over time, is the embedded script virus. Prior to the first observation of this class of viruses, viruses could be contracted only through e-mail by double clicking on an infected e-mail attachment. With the discovery of embedded script viruses,

such as the VBS_Kakworm and VBS_Bubbleboy, viruses can now infect a user's system when the e-mail is opened.

Mixed Application/Content-based Threats

Application-based wireless threats, in which an executable program carries some malicious code, affect the receiving device. The spread of this malicious code is slow since the user must download a program with malicious code and execute the program to become infected. At the other end of the spectrum are content-based threats that spread relatively benign text messages or generate cellular phone calls. Yet, these threats can spread rapidly due to the nature of their propagation medium—entire address books of e-mails.

The third type of threat is worse than the previous two types combined. While not yet seen in the wild or even in the laboratory, a threat that integrates techniques from both of these threat types could be formidable indeed. Imagine a virus that involved the unwitting download of sophisticated malicious code attached to a shareware program that wiped out wireless device applications and propagated itself rapidly across the wireless infrastructure via address books of e-mail. Such a virus could cause damage to each device it encountered and spread across a country, or across the world, overnight. Given the reality of the ILoveYou virus and its destructive power, without adequate comprehensive wireless infrastructure virus protection, some type of highly destructive, rapidly spreading wireless virus will inevitably surface.

Threats On The Horizon To Consumers And Corporations

In many parts of the world today, cellular phones are used almost exclusively for voice communication. Yet, as cellular phone technology is merged with the platform-independent Java programming language and emerging technologies such as Bluetooth, these cell phones will be able to send and receive data and applications, even from one wireless device directly to another wireless device. The line between PDAs and cellular phones is already blurred, and few dispute that the integrated, transaction-enabled wireless device that handles both voice and data will soon become a widespread reality.

So, as consumers download games that can be played offline, access the stock market, and pay for groceries with their wireless devices, business people will read e-mail, send short messages, and read graphics and charts on their wireless devices. Unfortunately, this wireless utopia is unlikely to come without a price—increasingly sophisticated wireless threats that utilise the same capabilities (e.g., connectivity, functionality, and speed). Viruses can spread from wireless device to wireless device, from wireless device to point-of-sale device (e.g., at the grocery counter), and from wireless device to PC.

The latter path offers a mode of transmission for viruses to wireless and wired internal LANs, and further propagation across the Internet. Currently, corporate IT managers have little control over which wireless and handheld devices their users are connecting to the network. Connecting a portable device (such as a PDA) into a PC that is connected (or subsequently connected) to the network is similar to inserting a floppy disk—that has not been scanned for viruses—into a computer.

A protection solution for the wireless infrastructure must have the following attributes:

❍ Multiple layers of protection to address the various entry points and transmission paths of viruses and malicious code
❍ Integration of centralised management of all antivirus solutions including maintenance of gateway, server, desktop, and device-level protection
❍ Implementation within the wireless infrastructure for early detection to minimise damage and costs
❍ Tools tailored to the wireless threat, rather than merely applying wired world tools
❍ Mechanisms for automatic maintenance, updating, and upgrading of virus protection since such protection is only as good as the last update
❍ Involve all parties via increased awareness of the potential threat including corporate IT managers, service providers, operating system and application developers, and end users

IM Viruses

Source: "Instant messaging safety and privacy tips"
available at
www.microsoft.com/athome/security/chat/imsafety.mspx

Instant messaging, commonly referred to as IM, is a method of online communication like e-mail. The main difference, as the name suggests, is that IM is instantaneous. Using an IM program—such as MSN Messenger, Windows Messenger, AOL Instant Messenger, Yahoo Messenger, or others—you and a friend can type messages to each other and see the messages almost immediately.

Because IM has become so popular, virus writers are using it to spread malicious programs. Read on to find out how to avoid getting or spreading a virus when you use IM.

Understanding Instant Message Viruses

Like e-mail viruses, instant message viruses are malicious or annoying programs that are designed to travel through IM. In most cases these viruses are spread when a person opens an infected file that was sent in an instant message that appeared to come from a friend.

When unsuspecting people open these files, their computers can become infected with a virus. Because of the virus, their computers may slow down or stop responding, or they may not notice any change at all. However, the virus might have installed a covert program on their computer that could damage software, hardware, or important files, and that may include spyware, which can track information entered on a computer.

A computer infected by a virus may continue to spread the infection by sending copies of the virus to everyone on your IM contact list. A contact list is the collection of IM names (similar to an e-mail address book) that you can store in your IM program.

Five Steps To Help Avoid Instant Message Viruses

As with most threats on the Internet, you can help keep yourself safe by taking basic precautions. If you know how to avoid e-mail

viruses, you'll already be familiar with many of these steps.

1. Be careful downloading files in IM. Never open, accept, or download a file in IM from someone you don't know. If the file comes from someone you do know, don't open it unless you know what the file is and you were expecting it. Contact the sender by e-mail, phone, or some other method to confirm that what they sent was not a virus.

2. Update your Windows software. Visit Windows Update to scan your computer and install any high-priority updates that are offered to you. If you have Automatic Updates enabled, the updates are delivered to you when they are released, but you have to make sure you install them.

3. Make sure you're using an updated version of your IM software. Using the most up-to-date version of your IM software can better protect your computer against viruses and spyware. If you're using MSN Messenger, install the updated version by visiting the MSN Messenger Web site and clicking the 'Download Now!' button.

4. Use anti-virus software and keep it updated. Anti-virus software can help to detect and remove IM viruses from your computer, but only if you keep the antivirus software current. If you've purchased a subscription from an anti-virus software company, your anti-virus software may update itself when you're connected to the Internet.

5. Use anti-spyware software and keep it updated. Some IM viruses may install spyware or other unwanted software on your computer. Anti-spyware software can help to protect your computer from spyware and remove any spyware you may already have. If you don't have anti-spyware software, you can download the new Microsoft Windows AntiSpyware (Beta) or another spyware removal tool.

Why You Need An E-mail Exploit Detection Engine

Source: "Why You Need an Email Exploit Detection Engine: Networks Must Supplement Anti-Virus Protection for Maximum Security"
available at
www.secinf.net/anti_virus/Why_You_Need_an_Email_Exploit_Detection_Engine_Networks_Must_Supplement_AntiVirus_Protection_for_Maximum_Security.html

Virus-writers are using increasingly complex and sophisticated techniques in their bid to circumvent anti-virus software and disseminate their viruses. A case in point was the notorious Nimda virus that used multiple methods to spread itself and was based on an exploit rather than on the virus/Trojan behaviour that anti-virus products typically search for. Anti-virus software, though essential, cannot combat such threats alone; an e-mail exploit detection tool is also necessary.

What Is An Exploit?

An exploit uses known vulnerabilities in applications or operating systems to execute a program or code. It "exploits" a feature of a program or the operating system for its own use, such as executing arbitrary machine code, read/write files on the hard disk, or gain illicit access.

What Is An E-mail Exploit?

An e-mail exploit is an exploit launched via e-mail. An e-mail exploit is essentially an exploit that can be embedded in an e-mail, and executed on the recipient's machine once the user either opens or receives the e-mail. This allows the hacker to bypass most firewalls and anti-virus products.

The Difference Between Anti-virus Software And E-mail Exploit Detection Software

Anti-virus software is designed to detect known malicious code. An e-mail exploit engine takes a different approach: it analyses the code for exploits that could be malicious. This means it can protect against new viruses, but most importantly against unknown viruses or malicious code. This is crucial as an unknown virus could be a one-off

piece of code, developed specifically to break into your network.

E-mail exploit detection software analyses e-mails for exploits—i.e., it scans for methods used to exploit the OS, e-mail client or Internet Explorer—that can permit execution of code or a program on the user's system. It does not check whether the program is malicious or not. It simply assumes there is a security risk if an e-mail is using an exploit in order to run a program or piece of code.

In this manner, an e-mail exploit engine works like an intrusion detection system (IDS) for e-mail. The e-mail exploit engine might cause more false positives, but it adds a new layer of security that is not available in a normal anti-virus package, simply because it uses a totally different way of securing e-mail.

Anti-virus engines do protect against some exploits but they do not check for all exploits or attacks. An exploit detection engine checks for all known exploits. Because the e-mail exploit engine is optimized for finding exploits in e-mail, it can therefore be more effective at this job than a general purpose anti-virus engine.

An Exploit Engine Requires Fewer Updates

An exploit engine needs to be updated less frequently than an anti-virus engine because it looks for a method rather than a specific virus. Although keeping exploit and anti-virus engines up-to-date involve very similar operations, the results are different. Once an exploit is identified and incorporated in an exploit engine, that engine can protect against any new virus that is based on a known exploit. That means the exploit engine will catch the virus even before the anti-virus vendor is aware of its emergence, and certainly before the anti-virus definition files have been updated to counter the attack. This is a critical advantage, as shown by the following examples that occurred in 2001.

The Lessons Of Nimda, BadTrans.B, Yaha And Bugbear

Nimda and BadTrans.B are two viruses that became highly known worldwide in 2001 because they infected a colossal number of Windows computers with Internet access. Nimda alone is estimated

to have affected about 8.3 million computer networks around the world, according to US research firm Computer Economics (November 2001). Nimda is a worm that uses multiple methods to automatically infect other computers. It can replicate through e-mail using an exploit that was made public months before Nimda hit, the MIME Header exploit. BadTrans.B is a mass-mailing worm that distributes itself using the MIME Header exploit. BadTrans.B first appeared after the Nimda outbreak. With their highly rapid infection rate, both Nimda and BadTrans.B took anti-virus vendors by surprise. Though the vendors tried to issue definition file updates as soon as they learned about each virus, the virus had already succeeded in infecting a large number of PCs by the time the anti-virus updates were released. Though both viruses used the same exploit, anti-virus vendors had to issue a separate definition file update for each. In contrast, an e-mail exploit detection engine would have recognized the exploit used and identified the attempt to automatically launch an executable file using the MIME header exploit. As a result, it would have blocked both worms automatically, preventing infection.

Other Examples Of Exploits

Double extension vulnerability viruses: Klez, Netsky and Lovegate.
What it does: Malicious files are given a double extension such as filename.txt.exe to trick the user into running the executable.

URL spoofing exploit viruses: No virus/worm has been found to be using this method. However it has been used to inject backdoors on Windows computers.
What it does: Allows spammers and phishers (scammers, or people trying to defraud computer users) to fool users to visit a malicious website instead of a legitimate one.

Object data file execution viruses: Bagle.Q.
What it does: Allows attackers to automatically infect unpatched versions of IE/Outlook (Express) by downloading and executing code from an HTTP site.

Computer Viruses In UNIX networks

Source: "Computer Viruses In Unix Networks"
Available at
www.cybersoft.com/whitepapers/papers/print/networks_print.html

The Existence Of The Problem And Its Nature

The problem of software attacks exists in all operating systems. These attacks follow different forms according to the function of the attack. In general, all forms of attack contain a method of self preservation which may be propagation or migration and a payload. The most common method of self preservation in Unix is obscurity. If the program has an obscure name or storage location, then it may avoid detection until after its payload has had the opportunity to execute. Computer worms preserve themselves by migration while computer viruses use propagation. Trojan horses, logic bombs and time bombs protect themselves by obscurity.

While the hostile algorithms that have captured the general public's imagination are viruses and worms, the more common direct problem on Unix systems are Trojan horses and time bombs. A Trojan horse is a program that appears to be something it is not. An example of a Trojan horse is a program that appears to be a calculator or other useful utility which has a hidden payload of inserting a back door onto its host system. A simple Trojan horse can be created by modifying any source code with the addition of a payload. One of the most favourite payloads observed in the wild is "/bin/rm -rf / >/dev/null 2>&1" This payload will attempt to remove all accessible files on the system as a background process with all messages redirected to waste disposal. Since system security is lax at many sites, there are normally thousands of files with permission bit settings of octal 777. All files on the system with this permission setting will be removed by this attack. Additionally, all files owned by the user, their group or anyone else on the system whose files are write accessible to the user will be removed. This payload is not limited to use by Trojan horses but can be utilized by any

form of attack. Typically, a time bomb can be created by using the "cron" or "at" utilities of the Unix system to execute this command directly at the specified time.

While the bin remove payload is a favourite of many authors, there are other traditional attacks which are not as overt in their destruction. These other attacks are more important because they bend the operation of the system to the purposes of the attacker while not revealing themselves to the system operator. Attacks of this form include the appending of an account record to the password file, copying the password file to an off site email address for leisurely cracking and modification of the operating system to include back doors or cause the transfer of money or property. It is extremely simple to email valuable information off site in such a manner as to insure that the recipient cannot be traced or located. Some of these methods are path dependent, however, the path selected is at the discretion of the attacker.

One of the most simple methods of inserting a back door is the well known suid bit shell attack. In this attack, a Trojanised program is used to copy a shell program to an accessible directory. The shell program is then set with permission bits that allow it to execute with the user id and permission of its creator. A simple one line suid bit shell attack can be created by adding the following command to a user's ".login" or any other file that they execute. Example: cp /bin/sh /tmp/gotu ; chmod 4777 /tmp/gotu

Trojan horses and time bombs can be located using the same methods required to locate viruses in the Unix environment. There are many technical reasons why these forms of attack are not desirable, the foremost being their immobility. A virus or worm attack is more important because these programs are mobile and can integrate themselves into the operating system. Of these two forms of attack, the virus attack is the hardest to detect and has the best chance of survival. Worms can be seen in the system process tables and eliminated since they exist as individual processes while virus attacks are

protected from this form of detection by their host programs. All of the methods used to detect and prevent viruses are also effective against the other forms of attack, therefore, the remainder of this paper will deal with the more serious problem of viral attacks.

Unix Virus Attacks

The promotion of the concept of "magical immunity" to computer viral attacks surfaces on a regular basis. This concept, while desirable, is misleading and dangerous since it tends to mask a real threat. Opponents of the possibility of viral attacks in Unix state that hardware instructions and operating system concepts such as supervisor mode or permission settings, security ratings like C2 or B1 provide protection. These ideas have been proven wrong by real life. The use of supervisor mode, the additional levels of protection provided by C2 and the mandatory access control provided by security level B1 are not necessary for viral activity and are therefore moot as a method of protection. This fact is supported by the existence of viruses that infect Unix systems as both scripts and binary.

In fact, virus attacks against Unix systems will eventually become more popular as simpler forms of attack become obsolete. Computer viruses have significantly more virility, methods of protection and opportunity for infection. Methods of protection have been highly refined in viruses, including rapid reproduction by infection, migration though evaluation of its environment, (boot viruses look for uninfected floppy diskettes) armor, stealth and polymorphism. In addition, the host system itself becomes a method of protection and propagation. Virus infected files are protected just as much by the operating system as are non-infected files. Introduction of viruses into systems have also been refined using technology called 'droppers'. A dropper is a Trojan horse that has a virus or viruses as a payload. Finally, extensive networking technology such as NFS (Network File System) allows viruses to migrate between systems without effort.

All of these reasons point to viruses as the future of hostile algorithms, however, the most significant reason for this determination is the effectiveness of the virus as a form of attack. Past experiments by Doctor Fred Cohen [1984] used a normal user account on a Unix system, without privileged access, and gained total security penetration in 30 minutes. Doctor Cohen repeated these results on many versions of Unix, including AT&T Secure Unix and over 20 commercial implementations of Unix. The results have been confirmed by independent researchers worldwide. Separate experiments by Tom Duff [1989] demonstrated the tenacity of Unix viruses even in the face of disinfectors. The virus used in Mr. Duff's experiment was a simple virus written in script. The virus was believed to have been reintroduced by the operating system from the automated backup and restore system. Re-infection took place after the system had been virus free for one year.

Heterogeneous Virus Attacks

Non-Unix PCs attached to a heterogeneous network that were infected with computer viruses originating from Unix servers have been observed. The Unix systems were not the original point of entry for the viruses. They were dormant while on the Unix systems but became harmful when they migrated to their target systems. The Unix systems acted as unaffected carriers of computer viruses for other platforms. For the sake of simplicity, I have named this effect after an historical medical problem of similar nature, 'Typhoid Mary Syndrome'. Networks and specifically Unix servers that provide network file systems are very susceptible to this problem.

This problem was first observed while investigating an infection of personal computers attached to a network with a large population of Unix servers and workstations. The virus was manually attacked on the personal computers using virus scanners. During the infection period all of the personal computers were disconnected from the network and idle. Once all the computers were disinfected, all removable media was tested and the infection was unobserved for a period of time, the

computers were reattached to the network. A few weeks later, a test of the computers using the same virus scanner indicated they had become re-infected with the same viruses. The source of infection was then identified as repositories of executables stored on the Unix file servers.

These repositories were organically grown centralized resources for all the personal computers because the Unix servers were effective at providing these shared services via NFS. In retrospect, this problem had to exist. The use of networked systems that were exported from the Unix platforms provided an easy, powerful method of transferring data, including executables. Some network designs provide all third party software from a network disk for ease of maintenance and reduced storage requirements. This easy access provides an open door for viruses.

Trans-platform Viruses Attack Unix

During late 1994 and early 1995 were observed multiple instances of at least three trans-platform virus attacks on Unix systems. All of these attacks involved MS-DOS viruses that attacked PC based Unix systems. The first attack involved a virus that corrupted the Unix file system every night. The attack was located using a virus scanner and indicated a Unix binary that was executed at midnight by cron. The MS-DOS virus had become embedded in the Unix executable where it was executed. The virus did not perform as designed in that the corruption was the result of the virus attempting to infect other files and was not an intended effect. The virus was reinstalled every morning when the system was restored. The second attack involved an MS-DOS virus that executed and was successful in infecting other files. Once again, the file system corrupted but it took longer in duration, thereby allowing the virus to propagate. The final infection involved a boot sector virus. Since this type of virus executes prior to the loading of the operating system , the differences between Unix and MS-DOS are moot. The PC-BIOS and processor chips are the same in both cases and the virus is able to execute according to design.

In fact, two different viruses were observed performing in this way. The first virus was spread by an MS-DOS setup diskette while the second virus was transmitted using a still undiscovered method. While we observed no boot sector infections of PC based Unix systems during 1994, we received reports from system administrators who were requesting information on our Unix anti-virus product because they had experienced hundreds of infections during 1995. In one instance, a single multinational company lost its entire international network overnight. The estimated cost in lost time, resources, and sales was in the millions of dollars.

Once it is understood that the BIOS and processor functions are the same for both operating systems, it is very easy to see how a trans-platform virus could be designed by intention. The virus would be able to process correctly by inspecting the operating system using only common BIOS calls and then modify its basic behaviour using a simple "if" structure.

Traditional Categories Of Protection And Their Failure

There are three traditional categories of protection, none of which provide complete or significant protection as stand-alone methods of implementation. The categories are Control, Inspection and Integrity. Each of these methods has traditionally been used separately.

Control has been the primary intent of the U.S. national standards on computer security. They deal with the control of access to the system, its functions, resources and the ability to move or share data in the system. These national standards are codified in a library generally referred to as the Rainbow series. (The name was given because the books have different colour covers making a library shelf look like a rainbow.) While these standards are a valuable and important aspect of computer security, they do not provide a deterrent against software attack. A virus is an effective way of gaining control over a system, even a highly controlled system such as a B1 rated version of Unix. In this case, control does not provide protection

against software attacks because of the viruses' ability to change permission sets with each new owner that is infected. A virus attack gains access to multiple users through shared files. Access control is designed to allow the sharing of files. The ability to share files is a basic need of the user and cannot be eliminated without destroying the usefulness of the system. Discretionary Access Control (DAC) is not protection against software attacks because it is a weak form of protection that can be bypassed and, as discretionary, is at the control of the end users who very often ignore it. Sites where the majority of the files on the system have no DAC protection are normal. (Many Unix sites have permission bit settings of 777 which allow anyone to read, write, execute or modify the file.) Mandatory Access Controls (MAC) also has little effect on virus activity for the same reasons, although MAC can be configured to be neither weak nor easy to bypass. Each time a virus attacks an executable file owned by a different user, it takes on the full privileges of that user, including access to files of other users whose permissions intersect the DAC and MAC permission sets of the infected user. On all systems, the need to share files forces the creation of users who exist in multiple permission sets. This multiple membership allows viruses to move between MAC compartments and levels. The reduction of multiple membership users will slow the advance of a virus but will not eliminate it. Finally, once a virus gains access to an operator account (root, operator, isso) it cannot be stopped by any form of control.

Inspection is the traditional way of locating both known holes in operating systems and in locating known viruses. The key word here is "known". System audit tools such as COPS, SATAN and others can only locate holes that are known to them. Virus scanners can only locate viruses that are known to them. This means that a virus scanner or inspection tool is obsolete even before it is shipped from the factory. It can only deal with the past, never the present or future since conditions searched for must exist at the time of coding. Virus scanner have to be constantly updated. This is becoming a problem with

the explosion of viruses being created by new authors and virus computer aided design and manufacturing tools (V-CAD/CAM).

It has been proposed that audit tools such as COPS can be used to deter virus infections because they strengthen the system's ability to control access and data movement. These inspection tools only improve control. As stated, control does not provide protection against virus attacks. It attempts to keep outside people out and inside people within their areas of authorization.

The third category of protection is Integrity. Integrity systems are intended to detect change. In the MS-DOS world, early integrity systems used cyclic redundancy character, CRC, values to detect change. A virus was then created which countered this protection. The virus determined the CRC value of the target file, infected it, and then padded the file until the CRC value computed the same. Many Unix users still use this method of change detection, or worse, they attempt to use the date of last modification as an indication of change. The date of last modification can be changed to any value on Unix systems with a simple user command. On many systems an option of the "touch" command provides this ability.

Any integrity tool that does not use cryptographic methods is of little value. In fact, if the integrity system fails to detect critical changes, then the false sense of security created in the system operator can be devastating to the system. An integrity tool, CIT, was created using the RSA Associates MD5 cryptographic hash algorithm. Since the algorithm is cryptographic, it can detect even a single bit flip and cannot be misled by any known means. In addition, during the development of CIT, it was determined that it was necessary to detect additions and deletions to the file system since these could be indications of non-infectious attacks such as performed by Trojan horses, worms and hackers. In this way, a rolling baseline can be created that will allow the system operator to quickly recover from any form of file system attack. Modifications to the protected

file system created by unauthorized users or software attacks can be detected and removed. Using a tool of this type allows the administrator to locate the approximate time of attack since the modification will have taken place between two known timed events, the last and current execution of the integrity tool. Finally, integrity tools can be used to determine if a third party file has been modified or tampered with prior to use. Some manufacturers of Unix operating systems now publish MD5 digests of their systems. Using these digests, it is possible to determine that the file on your system is exactly as it should be. There was no degradation from misreading the installation media, deterioration of the disk system or intentional modification. If a manufacturer does not publish a list, then end users can create their own by installing an operating system on multiple systems from different media sources. The created digests of each system should agree.

Non-traditional Categories Of Protection And Their Failure

In the past, fencing systems were sold as a popular method of virus protection on PC platforms. A fencing system write protects parts of the disk using a hardware board that is added to the system bus. Since a virus cannot infect a file that is write protected using hardware, it appears to be a good method. The obvious drawback is that the user cannot write to the disk if it is write protected. The fencing system therefore had to create zones of protection so that the user could perform useful work. Viruses happily infected the unprotected zones. Fencing systems appear to have never been marketed for Unix systems.

Projection Of Future Problems

The problem of attack software written for and targeted against Unix systems will continue to grow, especially now that the Internet has gained popularity. Unix systems are the backbone of the world wide Internet. Viruses will become more prevalent because they provide all of the benefits of other forms of attack while having few drawbacks. Trans-platform viruses may become common as an effective attack. All of the methods cur-

rently used in creating MS-DOS viruses can be ported to Unix. This includes the creation of automated CAD/CAM virus tools, stealth, polymorphism and armour. The future of viruses on Unix is already hinted at by the wide spread use of Bots and Kill-Bots, (slang term referring to software robots). These programs are able to move from system to system performing their function. Using a Bot as a dropper or creating a virus that includes bot-like capability is simple. With the advent of global networks, the edge between viruses, bots, worms and Trojans will blur. Attacks will be created that use abilities from all of these forms and others to be developed. There have already been cases where people have used audit tools such as COPS and SATAN to attack a system. Combining these tools with a virus CAD/CAM program will allow a fully functional virus factory to create custom viruses and attacks against specific targets such as companies that are disliked by the propitiator. The information services provided by the Internet already provide sufficient information in the form of IP addresses and email domain addresses to identify, locate and attack systems owned by specific entities.

Finally, viruses and worms can provide the perfect format for a hostage shielded denial of service attack. It is well known that an Internet attached system can be made to "disappear" or crash by flooding it with IP packets. Site administrators can protect their systems from crashing by programming their local router to filter out packets from the attacking source. The system will still disappear because legitimate users will be squeezed out by the flood of attack packets, but filtering at the router can at least save the system from crashing. Unfortunately, anyone can masquerade as someone else on the Internet by merely using their IP address. This attack can send a barrage of packets to the target site, each of which has a different source IP address. It is not possible to use a router to filter from this type of attack, but the Internet service provider can trace the source of attack by physical channel without relying upon the IP address. In cooperation with other Internet providers, the attacker can be isolated from the Internet for a

short time. Hopefully, the attacker will become bored and go away or can be identified for action by law enforcement. Another possibility is to use viruses to generate the attack. If a virus is successful in spreading to thousands of sites on the Internet and is programmed to start an IP attack against a specific target on the same day at the same time then there is no way to stop the attack because it has originated from thousands of sites all of which are live hostages. The site under attack will have to go off line since the Internet service providers will be helpless in the face of a coordinated dispersed attack. Since the impact against each individual hostage system is low, the hostages may not even notice that there is a problem. The Internet service provider attached to the target system is in the best position to detect the attack, however, they are as subject to this attack as the target since they may 'crash' from the excessive bandwidth usage flooding their network from multiple sources.

Scenario Of A Virus Attack Against A Secure Unix Network

The military and many other companies believe that they are protected against focused attacks because they employ a closed network configuration. In some cases these networks may also use highly secure 'B' rated operating systems [NCSC-TG-006]. Typically, the network will not allow modems, Internet connections or have any electronic connections to organizations outside of the immediate need. In addition, the networks are almost always heterogeneous because of legacy equipment, primarily PC systems. The network designers normally allow the PC systems to retain their floppy disk drives even thought their attachment to a network renders them nonessential. Networks of this type have been considered secure, however, they are open to information warfare attacks via a focused virus. Assuming that the propitiator is an outsider without access to the equipment or premises, one possible method of attack against this type of network would take advantage of both the Typhoid Mary Syndrome and Trans-platform Viruses to produce an attack that is targeted against the Unix systems but origi-

nated from an attached PC. A virus can be created whose pay-load is triggered by executing on a PC that is attached to the target network. This is not hard with a little inside information about the configuration of the network. The propitiator would then install the virus at all of the local Universities in the hope that someone working at the installation is taking a night class or that one of their children will unknowingly infect a common usage home computer. At that point, the virus has a good chance of entering the target network. This is a well known vector and is enhanced because the virus will not reveal itself. Once on the target system, the PC virus will act like a dropper releasing a Unix virus into the backbone. The payload virus may be necessary because many Unix backbone systems are not PC compatible. The Unix virus payload can then install a back door which can be remotely directed. In addition, the virus can create a covert channel by making use of messenger viruses. While the use of messenger viruses are slow and have low bandwidth, they are bidirectional and can be used for command and control of more complex attacks.

Conclusion

The problem of attack software targeted against Unix systems will continue to grow. Viruses may become more prevalent because they provide all of the benefits of other forms of attack, while having few drawbacks. Trans-platform viruses may become common as an effective attack. All of the methods currently used in creating MS-DOS viruses can be ported to Unix. This includes the creation of automated CAD/CAM virus tools, stealth, polymorphism and armour. The future of viruses on Unix is already hinted at by the wide spread use of Bots and Kill-bots (slang term referring to software robots). These programs are able to move from system to system performing their function. Using a Bot as a dropper or creating a virus that includes bot like capability is simple. With the advent of global networks, the edge between viruses, bots, worms and Trojans will blur. Attacks will be created that use abilities from all of these forms and others to be developed. There have already been cases where people have used audit tools such as COPS

and SATAN to attack a system. Combining these tools with a virus CAD/CAM program will allow a fully functional virus factory to create custom viruses to attack specific targets.

As these problems unfold, new methods of protection must be created. Research has hinted at several promising methods of protection, including real time security monitors that use artificial intelligence for simple decision making.

Even with the current problems and the promise of more sophisticated problems and solutions in the future, the one thing that is believed to be certain is that Unix or Unix-like systems will continue to provide a pay back that is well worth the cost of operating them.

Linux Network Security

Network security is becoming more and more important as people spend more and more time connected. Compromising network security is often much easier than compromising physical or local security, and is much more common.

There are a number of good tools to assist with network security, and more and more of them are shipping with Linux distributions.

Packet Sniffers

One of the most common ways intruders gain access to more systems on your network is by employing a packet sniffer on a already compromised host. This "sniffer" just listens on the Ethernet port for things like passwd and login and su in the packet stream and then logs the traffic after that. This way, attackers gain passwords for systems they are not even attempting to break into. Clear-text passwords are very vulnerable to this attack.

Example: Host A has been compromised. Attacker installs a sniffer. Sniffer picks up admin logging into Host B from Host C. It gets the admins personal password as they login to B. Then, the admin does a su to fix a problem. They now have the root password for Host B. Later the admin lets someone telnet from his account to Host Z on another site. Now the attacker has a password/login on Host Z.

In this day and age, the attacker doesn't even need to compromise a system to do this: they could also bring a laptop or pc into a building and tap into your net.

Using ssh or other encrypted password methods thwarts this attack. Things like APOP for POP accounts also prevents

this attack. (Normal POP logins are very vulnerable to this, as is anything that sends clear-text passwords over the network.)

SATAN, ISS, and Other Network Scanners

There are a number of different software packages out there that do port and service-based scanning of machines or networks. SATAN, ISS, SAINT, and Nessus are some of the more well-known ones. This software connects to the target machine (or all the target machines on a network) on all the ports they can, and try to determine what service is running there. Based on this information, you can tell if the machine is vulnerable to a specific exploit on that server.

SATAN (Security Administrator's Tool for Analyzing Networks) is a port scanner with a web interface. It can be configured to do light, medium, or strong checks on a machine or a network of machines. It's a good idea to get SATAN and scan your machine or network, and fix the problems it finds. Make sure you get the copy of SATAN from metalab or a reputable FTP or web site. There was a Trojan copy of SATAN that was distributed out on the net. http://www.trouble.org/~zen/satan/satan.html. Note that SATAN has not been updated in quite a while, and some of the other tools below might do a better job.

ISS (Internet Security Scanner) is another port-based scanner. It is faster than Satan, and thus might be better for large networks. However, SATAN tends to provide more information.

Abacus is a suite of tools to provide host-based security and intrusion detection. Look at it's home page on the web for more information. http://www.psionic.com/abacus/

SAINT is a updated version of SATAN. It is web-based and has many more up-to-date tests than SATAN. You can find out more about it at: http://www.wwdsi.com/~saint

Nessus is a free security scanner. It has a GTK graphical

interface for ease of use. It is also designed with a very nice plug in setup for new port-scanning tests. For more information, take a look at: http://www.nessus.org

Denial of Service Attacks

Denial of service attacks have increased greatly in recent years. Some of the more popular and recent ones are listed below. Note that new ones show up all the time, so this is just a few examples. Read the Linux security lists and the bugtraq list and archives for more current information.

❍ SYN Flooding - SYN flooding is a network denial of service attack. It takes advantage of a "loophole" in the way TCP connections are created. The newer Linux kernels (2.0.30 and up) have several configurable options to prevent SYN flood attacks from denying people access to your machine or services. See Section 7 for proper kernel protection options.

❍ Pentium "F00F" Bug - It was recently discovered that a series of assembly codes sent to a genuine Intel Pentium processor would reboot the machine. This affects every machine with a Pentium processor (not clones, not Pentium Pro or PII), no matter what operating system it's running. Linux kernels 2.0.32 and up contain a work around for this bug, preventing it from locking your machine. Kernel 2.0.33 has an improved version of the kernel fix, and is suggested over 2.0.32. If you are running on a Pentium, you should upgrade now!

❍ Ping Flooding - Ping flooding is a simple brute-force denial of service attack. The attacker sends a "flood" of ICMP packets to your machine. If they are doing this from a host with better bandwidth than yours, your machine will be unable to send anything on the network. A variation on this attack, called "smurfing", sends ICMP packets to a host with your machine's return IP, allowing them to flood you less detectably. You can find more information about the "smurf" attack at http://www.quadrunner.com/~chuegen/smurf.txt

If you are ever under a ping flood attack, use a tool like tcp-dump to determine where the packets are coming from (or appear to be coming from), then contact your provider with this information. Ping floods can most easily be stopped at the router level or by using a firewall.

❍ Ping o' Death - The Ping o' Death attack sends ICMP ECHO REQUEST packets that are too large to fit in the kernel data structures intended to store them. Because sending a single, large (65,510 bytes) "ping" packet to many systems will cause them to hang or even crash, this problem was quickly dubbed the "Ping o' Death." This one has long been fixed, and is no longer anything to worry about.

❍ Teardrop / New Tear - One of the most recent exploits involves a bug present in the IP fragmentation code on Linux and Windows platforms. It is fixed in kernel version 2.0.33, and does not require selecting any kernel compile-time options to utilize the fix. Linux is apparently not vulnerable to the "newtear" exploit.

You can find code for most exploits, and a more in-depth description of how they work, at http://www.rootshell.com using their search engine.

VPNs - Virtual Private Networks

VPN's are a way to establish a "virtual" network on top of some already-existing network. This virtual network often is encrypted and passes traffic only to and from some known entities that have joined the network. VPNs are often used to connect someone working at home over the public Internet to an internal company network.

If you are running a Linux masquerading firewall and need to pass MS PPTP (Microsoft's VPN point-to-point product) packets, there is a Linux kernel patch out to do just that. See: ip-masq-vpn.

There are several Linux VPN solutions available:

❍ vpnd. See the http://sunsite.dk/vpnd/.
❍ Free S/Wan, available at http://www.xs4all.nl/~freeswan/
❍ ssh can be used to construct a VPN. See the VPN mini-howto for more information.
❍ vps (virtual private server) at http://www.strongcrypto.com.
❍ yawipin at http://yavipin.sourceforge.net
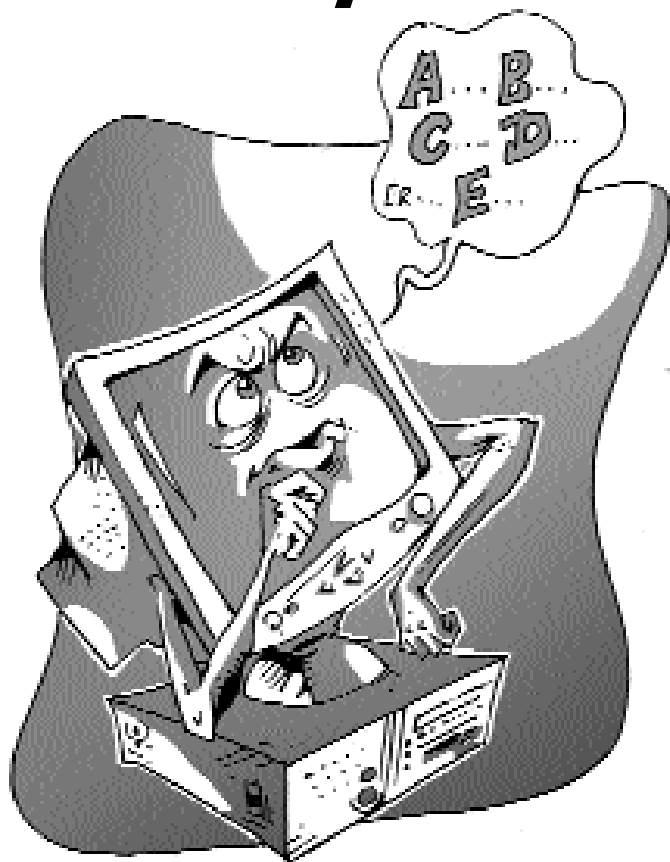
Glossary

A comprehensive glossary for your ready reference whenever you want to crack the technical mumbo-jumbo pertaining to virus or anti-virus technologies.

Adware or Ad ware

This is software that downloads and displays advertisements. This kind of software is often bundled with Freeware.

Alias

There is no standard, accepted rule for naming viruses. Hence, even though informal groups, such as CARO, have discussed conventions for virus naming, differences still exist between antivirus software companies and research organisations. Thus, where the term 'alias' or 'also known as' occurs, it refers to different names that the same virus may have been given by other sources.

Annoyance

Any trojan that does not cause any major damage, but instead annoys a user by turning the text on the screen upside down, or making mouse motions erratic, and so on.

ANSI Bomb

Character sequences that reprogram specific keys on the keyboard. If ANSI.SYS is loaded, some bombs will display colourful messages, or have interesting (but unwanted) graphical effects.

Anti-emulation

To reliably detect polymorphic viruses, scanners include code emulators to simulate the running of executable code and check whether it decrypts to a known virus. An emulator must stop emulating a program once it is no longer necessary, and for performance reasons many emulators have simple rules for quickly determining a stopping point. Some polymorphic viruses include tricks attempting to defeat these code emulators by fooling them into quitting the emulation before the decryption code has finished its work. Such methods are commonly called anti-emulation techniques.

Antivirus Virus

The idea of making an antivirus program viral so that it can propagate to where it is most needed is a very old one. Such a program would be an antivirus virus. It is universally agreed among rep-

utable antivirus researchers to be a very bad, even dangerous, idea, and should be avoided at all costs.

Anti-heuristic

Anti-heuristic techniques are efforts by virus writers to avoid their code being detected as a possible new virus by heuristic detection. What works depends on heuristics approaches of different scanners, but some code obfuscation techniques appear clearly anti-heuristic.

Appender

A virus that inserts a copy of its code at the end of its victim file is known as an appender or appending virus. (c.f. Cavity Infector, Companion Virus, Overwriter, Prepender)

Armoured Virus

Viruses that use special tricks to make tracing them in a debugger and/or disassembling them difficult are said to be 'armoured'. The purpose of armouring is primarily to hinder virus analysts reaching a complete understanding of the virus' code. An early example of an armoured virus is Whale.

AV Killer

Any hacker tool intended to disable a user's anti-virus software to help elude detection. Some will also disable personal firewalls.

Backdoor

A program that surreptitiously allows access to a computer's resources (files, network connections, configuration information etc.) via a network connection is known as a backdoor or remote access trojan. Note that such functionality is often included in legitimate software designed and intended to allow such access.

Bait File, Goat File, Decoy File

Some generic approaches to virus detection create 'dummy' program files which are written to the drives of the machines being monitored. These files are regularly checked for modification, or created, checked and then deleted. Such files are sometimes called 'goat files', 'decoy files' or 'bait files' because they are not intend-

ed to be run for any practicable purpose, and act solely as 'bait' to trap and detect the presence of an active virus.

Bimorphic Virus

An encrypted virus that has two forms of the decryption code, usually randomly selecting between them when writing its decryptor to a new replicant.

Binder

A tool that combines two or more files into a single file, usually for the purpose of hiding one of them.

BIOS

Basic Input/Output System is the lowest level program in a PC, which provides an interface with the PC's hardware. A PC's BIOS is also responsible for initiating the operating system bootstrap process by loading the boot sector of a diskette or the master boot record of a hard drive and passing control to it.

ActiveX

ActiveX controls are software modules based on Microsoft's Component Object Model (COM) architecture. On the Internet, ActiveX controls can be linked to Web pages and downloaded by an ActiveX-compliant browser. ActiveX controls turn Web pages into software pages that perform like any other program launched from a server, and can have full system access. In most instances this access is legitimate, but one should be cautious of malicious ActiveX applications.

Attachments

Attachments are files added to an outgoing e-mail. In Microsoft e-mail clients, e-mail carrying an attachment will have a paper-clip type icon beside the description. Also in Microsoft e-mail clients, an icon representing the file type will be embedded at the end of the body of the e-mail message. Attachments have become a known harbinger of virus infection. Virus authors and distributors often give the file a double extension. Users who do not have proper viewing settings configured in Internet Explorer

can be tricked into believing an executable file is a benign bitmap, or graphic, file. To prevent this, ensure file extension viewing is enabled on the system.

Antivirus

Antivirus refers to the products and technology used to detect malicious code, prevent it from infecting your system, and remove malicious code that has infected the system. Typically, antivirus vendors share information and resources to ensure rapid response to malicious code outbreaks. Most antivirus vendors participate in independent testing which certifies their products to detect or disinfect viruses.

Applet

Any miniature application transported over the Internet, especially as an enhancement to a Web page. Authors often embed applets within the HTML page as a foreign program type.

Attack

An attempt to subvert or bypass a system's security. Attacks may be passive or active. Active attacks attempt to alter or destroy data. Passive attacks try to intercept or read data without changing it. See Also: Brute Force Attack, Denial of Service, Hijacking, Password Attacks, Password Sniffing

Attributes

Characteristics assigned to all files and directories. Attributes include: Read Only, Archive, Hidden or System.

Background Scanning

A feature in some antivirus software to automatically scan files and documents as they are created, opened, closed or executed.

Backup

The process of creating duplicate data. Some programs backup data files while maintaining both the current version and the preceding version on disk. However, a backup is not considered secure unless it is stored away from the original.

Boot Code

The program recorded in a boot sector is known as boot code. Boot sectors usually contain boot code because these small programs have the job of starting to load a PC's operating system once the BIOS completes its POST checks.

Boot Infector, Boot Sector Infector, BSI

A boot sector infector virus places its starting code in the boot sector. When the computer tries to read and execute the program in the boot sector, the virus is loaded into the memory, where it can gain control over basic computer operations. From the memory, a boot sector infector can spread to other drives (floppy, network etc.) on the system. Once the virus is running, it usually executes the normal boot program, which it stores elsewhere on the disk.

Boot Record

The program recorded in the boot sector. This record contains information on the characteristics and contents of the disk and information needed to boot the computer. If a user boots a PC with a floppy disk, the system reads the boot record from that disk.

Boot Sector

An area located on the first track of floppy disks and logical disks that contain the boot record. Boot sector usually refers to this specific sector of a floppy disk, whereas the term Master Boot Sector usually refers to the same section of a hard disk. See Also: Master Boot Record

Boot Virus

A virus that infects boot sectors. Also, refer to Boot Sector Infector for more details.

Bug

An unintentional fault in a program that causes actions neither the user nor the program author intended.

CARO

It stands for Computer Antivirus Research Organisation, which is an informal group of professional antivirus researchers committed to improving the state of the art.

Cavity Infector, Cavity Virus

A cavity virus overwrites a part of its host file without increasing the length of the file while also preserving the host's functionality.

Class Infector

A class infector is a macro virus whose code resides in one or more class modules.

Cluster Virus, Link virus

Apart from directly infecting host files as appenders and prependers do, there are other ways to intercept calls to an executable file and run malicious code, either before or instead of, the code from the intended file. One such method is cluster infection, used by a small number of DOS viruses.

CMOS

Complementary Metal Oxide Semiconductor: The battery backed RAM used in AT and later PCs to store hardware configuration information uses CMOS technology. As this memory is not in the CPU address space, but addressed via I/O port reads and writes, its contents cannot be directly executed. This means that viruses cannot reside in nor infect the CMOS RAM. Some viruses alter the contents of the CMOS RAM as a payload, either scrambling them or removing the reference to the floppy drive so the hard drive's (infected) MBR will always run first during boot-up.

Commercial RAT

Any commercial product that is normally used for remote administration, but which might be exploited to do this without a user's consent or awareness is called a Commercial RAT. Also see RAT.

Companion Virus

There are more methods of infecting a system other than the most commonly used one of modifying an existing file (see Parasitic Virus). Given the way command-line interpreters (or shells) of several operating systems work, a virus can copy itself onto the system as an entire program yet be sure that much of the time, attempts to invoke a program will result in the virus' code being run first. Such programs are known as companion viruses and there are several forms of this infection method.

Constructor Kit, Generator Kit

Some virus writers are not content with writing their own viruses and have wondered about bringing the 'opportunity' of becoming a virus writer to the masses. The solution to this is usually some form of 'construction kit'—a program even a non-programmer can run, feed some parameters into and then produce a virus.

Crack

Any software designed to modify other software for the purpose of removing usage restrictions. An example is a 'patcher' or 'patch generator' that will replace bytes at specified locations in a file, rendering it as good as a fully-licensed version.

Data Diddlers

Data Diddlers is a popular name for a virus that contains a data modifying payload. This type of virus may, for instance, change 0s to 9s in an MS Excel spreadsheet or it may even replace certain words.

Checksum

An identifying number calculated from file characteristics. The slightest change in a file changes its checksum. This is used to ensure that you have the exact same file as the one written by the author.

Clean, Disinfect

To remove a virus or other malicious software from a computer, file or disk.

COM File

A type of executable file that is limited to 64 KB. These simple files are often used for utility programs and small routines. Because COM files are executable, viruses can infect them. This file type has the extension COM.

Cookie

Cookies are blocks of text placed in a file on your computer's hard disk. Web sites use cookies to identify users who revisit the site. Cookies might contain login or registration information, "shopping cart" information or user preferences. When a server receives a browser request that includes a cookie, the server can use the information stored in the cookie to customise the Web site for the user. Cookies can be used to gather more information about a user than would be possible without them.

DDoS

Distributed Denial of Service. Attempts to DoS large sites using most forms of resource exhaustion attack, and particularly network bandwidth wasting strategies, are often impossible for a single attacking machine because of the sheer scale of resources available to the attacked site.

Denial of Service, DoS

An attack on a computer system intended to reduce, or entirely block, the level of service that 'legitimate clients' can receive from that system. These range in scope from network bandwidth wasting and/or swamping through exhausting various machine resources (such as memory, disk space, thread or process handles) required by the process(es) providing the service. They usually work by exploiting vulnerabilities that eventually crash the service process or the underlying system. Although not commonly associated with viruses, denial of service components are included in some viral payload routines.

Destructiveness

This is measured based on the amount of damage that a malicious program can possibly achieve once a computer has been infected.

These metrics can include attacks to important operating system files, triggered events, clogging e-mail servers, deleting or modifying files, releasing confidential information, performance degradation, compromising security settings, and the ease with which the damage may be fixed.

Dialler

Software that dials a phone number. Some dialers connect to local Internet Service Providers and are beneficial as configured. Others connect to toll numbers without user awareness or permission.

Direct Action Virus

A virus that attempts to locate and infect one or more targets when it is run and then exits, is called a direct action virus. In single-tasking operating systems such as DOS, direct action viruses usually only infect a small number of targets during each run, as the 'find then infect' process slows the normal execution of the infected host from which the virus is running and significant slowing of a machine is likely to warn its user of the presence of something 'untoward'.

DOS

Disk Operating System-most famously, MS DOS and IBM DOS, but also DR DOS and others.

Downloader

A downloader is a program that automatically downloads and runs and/or installs other software without the user's knowledge or permission.

Dropper, Injector

A program that installs a virus, but is not, itself, infected is known as a dropper. These are not very common and most are used to install boot viruses.

Disinfection

Most anti-virus software carries out disinfection after reporting the presence of a virus. During disinfection, the virus may be removed and, whenever possible, any affected data is recovered.

DOC File

A Microsoft Word Document File. In the past, these files contained only document data, but with many newer versions of Microsoft Word, DOC files also include small programs called macros. Many virus authors use the macro programming language to associate macros with DOC files. This file type has the extension DOC.

EEPROM

Electrically Erasable and Programmable Read-Only Memory. A type of ROM whose contents are non-volatile, but modifiable through the application of appropriate chip reprogramming voltages.

EICAR

European Institute for Computer Antivirus Research. A group of academics, researchers, law enforcement specialists and other technologists united against writing and proliferation of malicious code such as computer viruses or trojan horses, and against computer crime, fraud and the misuse of computers or networks

E-mail Worm

A commonly used misnomer for mass mailing viruses

Emulator

A commonly used method for detecting polymorphic viruses is to simulate running part of a program's code in an emulator. The purpose is to see if the code decrypts known virus code. There are several essentially irresolvable issues with emulator design. For example, ensuring they don't run for 'too long' on each file thus slowing the scanner down, and making them complex enough to include sufficient aspects of the environment they simulate that anti-emulation and emulation detection techniques employed in some viruses do not reduce their usefulness.

Encrypted Virus

An early attempt at evading scan string driven virus detectors was self-encryption with a variable key.

Encryption Tool

Any software that can be used to scramble documents, software, or systems so that only those possessing a valid key are able to unscramble it. Encryption tools are used to secure information; sometimes unauthorised use of encryption tools in an organisation is a cause for concern.

EPROM
Erasable and Programmable Read-Only Memory. A type of ROM whose contents are non-volatile but modifiable through the application of appropriate chip reprogramming voltages.

Error Hijacker
Any software that resets your browser's settings to display a new error page when a requested URL is not found. Hijacks may reroute your information and address requests through an unseen site, capturing that info. In such hijacks, your browser may behave normally, but be slower.

Exploit
A way of breaking into a system. An exploit takes advantage of a weakness in a system in order to hack it. Exploits are the root of the hacker culture.

EXE File
An executable file; as contrasted with a document or data file. Usually, executed by double-clicking its icon or a shortcut on the desktop, or by entering the name of the program at a command prompt. Executable files can also be executed from other programs, batch files or various script files.

False Positive, False Negative
These terms derive from their use in statistics. If it is claimed that a file or boot sector is infected by a virus when in reality it is clean, a false positive (or Type-I) error is said to have occurred. Conversely, if a file or boot sector that is infected is claimed to not be infected, a false negative (or Type-II) error has been made. From

an antivirus perspective, false negatives probably seem more seri-
ous than false positives, but both are undesirable.

Fast Infector

When programs infected with common file infectors are run, the
virus code usually gets control first. It then checks it has not
already gone resident, copies itself into memory, and hooks a sys-
tem interrupt or event handler associated with the host platform's
'load and execute' function. When that function is subsequently
called, the virus' infection routine runs, checking whether the
program that is about to run has been infected already, and if
not, infects it.

Mass Mailer, Fast Mailer

A virus that distributes itself via e-mail to multiple addressees at
once is known as a mass mailer.

FAT, File Allocation Table

A crucial part of the standard file systems employed in all versions
of DOS and Windows 9x. The FAT records the chaining of disk clus-
ters and the final cluster in a file. A file's first cluster is stored in
its directory entry and also acts as an offset into the FAT's chain-
ing table so the rest of the file can be located.

Field Sample, Field Virus, In the Field

Sometimes viruses are said to be 'in the field' or 'reported from
the field'. This may be loose usage of the term, or it may be to draw
the distinction between viruses that have been seen in a small
number of real-world infection incidents ('in the field') and those
that have reached the top half of the WildList ('In the Wild').

File Infector

These are viruses that attach themselves to (or replace; see
Companion Virus) .COM and .EXE files, although in some cases
they will infect files with other extensions such as .SYS, .DRV, .BIN,
.OVL, .CPL, .DLL, .SCR and others. The most common file viruses are
resident viruses, loading into memory at the time the first copy is
run, and taking clandestine control of the computer. Such viruses

commonly infect additional program files as they are run or even just accessed. But there are many non-resident viruses, too, which simply infect one or more files whenever an infected file is run.

File Race Condition

Some applications store information in unsecured files and folders like the temp directory. A file race condition occurs where an attacker has the chance to modify these files before the original application has finished with them. If the attacker successfully monitors, attacks and edits these temp files, the original application will then process them as if they were legitimate. The name of this kind of attack is from the attackers 'race to edit the file'.

Firewall Killer

Any hacker tool intended to disable a user's personal firewall. Some will also disable resident anti-virus software.

Flash Memory

Flash memory became of interest to antivirus researchers when the full measure of CIH's payload was decoded. Because the BIOS of most Pentium-class and later PCs is shipped on a flash memory chip and most mainboard and system designs result in write-mode for that memory being readily enabled, the BIOS of a PC can no longer be considered 'carved in stone'.

Flooder

A program that overloads a connection by any mechanism, such as fast pinging, causing a DoS attack.

FTP Server

When installed without user awareness, an FTP server allows an attacker to download any file in the user's machine, to upload new files to that machine, and to replace any existing file with an uploaded file.

FDISK /MBR

If you have MS-DOS version 5.0 or later, the command "FDISK /MBR" can remove viruses which infect the master boot sector but

do not encrypt it. Using this command can produce unexpected results and cause unrecoverable damage.

Firewall

A firewall prevents computers on a network from communicating directly with external computer systems. A firewall typically consists of a computer that acts as a barrier through which all information passing between the networks and the external systems must travel. The firewall software analyses information passing between the two and rejects it if it does not conform to pre-configured rules.

Germ

A first generation sample of a virus. Technically, the term is reserved for forms of the virus that are in some way 'special', such that another sample could not be produced as the result of a normal infection event.

Ghost Positive

This is a specific form of false positive, in which the error is due to 'leftover pieces' or 'remnants' of a virus that are incorrectly detected and reported as an infection. As the virus is not present or present but inactive, it is erroneous for a scanner to report an infection.

Globbing

Globbing is the use of wildcard characters or arguments to greatly increase the amount of data requested. An example is "dir *.*" in DOS, this command is asking for all file names with all file extensions (everything) in the current directory. By making globbing requests to a Web server it is sometimes possible to cause a Denial of Service attack as the server is too busy to deal with legitimate requests.

Heuristic Analysis, Heuristic Scan

This is Behaviour-based analysis of a computer program by anti-virus software to identify a potential virus. Often heuristic scanning produces false alarms when a clean program might behave as a virus.

Hijacking

This is an attack whereby an active, established, session is intercepted and used by the attacker. Hijacking can occur locally if, for example, a legitimate user leaves a computer unprotected. Remote hijacking can occur via the Internet.

Hoax

A hoax is a message, typically distributed via e-mail or newsgroups, which is written to deliberately spread fear, uncertainty and doubt. Just like the viruses they purport to describe, they are sent from user(s), slowing network and Internet traffic and causing damage 'per se', by wasting users time and by prompting well meaning, and unnecessary clean up procedures. However, these messages may be regarding completely fictitious viruses and trojans, or they may even be misleadingly warning users about legitimate programs.

Homepage Hijacker

Any software that changes your browser's home page to some other site without your explicit permission. Hijacks may reroute your info and address requests through an unseen site, capturing that info. In such hijacks, your browser may behave normally, but be slower.

Hostile ActiveX

An ActiveX control is essentially a Windows program that can be distributed from a Web page. These controls can do literally anything a Windows program can do. A Hostile ActiveX program does something that its user did not intend for it to do, such as erasing a hard drive, dropping a virus or trojan into your machine, or scanning your drive for tax records or documents. As with other Trojans, a Hostile ActiveX control will normally appear to have some other function than what it actually has.

Hostile Java

Browsers include a "virtual machine" that encapsulates the Java program and prevents it from accessing your local machine. The

theory behind this is that a Java "applet" is really content such as graphics, rather than full application software. However, as of July, 2000, all known browsers have had bugs in their Java virtual machines that would allow hostile applets to break out of this sandbox and also access other parts of the system. As a matter of fact, most security experts browse with Java disabled on their computers, or encapsulate it with further sandboxes/virtual-machines.

Hostile Script

A script is a text file with a .VBS, .WSH, .JS, .HTA, .JSE, .VBE extension that is executed by Microsoft WScript or Microsoft Scripting Host Application, interpreting the instructions in the script and acting on them. A hostile script performs unwanted actions.

HTTP Server

When installed without user awareness, an HTTP server allows an attacker to use a Web browser to view and thus retrieve information collected by other software placed in the user's machine.

Hole

Vulnerability in the design software and/or hardware that allows circumvention of security measures.

Host

A term often used to describe the computer file to which a virus attaches itself. Most viruses run when the computer or user tries to execute the host file.

Impact

The extent to which an attacker may gain access to a system and the severity of it on the organisation.

IRC War

Any tool that uses Internet Relay Chat for spoofing, eavesdropping, sniffing, spamming, breaking passwords, harassment, fraud, forgery, electronic trespassing, tampering, hacking, nuking, system contamination including without limitation use of

viruses, worms and Trojan horses causing unauthorised, damaging or harmful access or retrieval of information and data on your computer and other forms of activity that may even be considered unlawful.

In The Wild

A virus is "in the wild" if it is verified as having caused an infection outside a laboratory situation. Most viruses are in the wild and differ only in prevalence.

Joiner

Loosely a joiner is a program that takes two or more files and 'sticks them together'. In antivirus and malware circles it is typically used in reference to utilities that join two or more files together with one or more of these being executables.

Joke Program

In general, they aim to entertain either the recipient or the supplier of the program, although it is probably the case that the joke is usually at the expense of the recipient.

Infection

The action a virus carries out when it enters a computer system or storage device.

JavaScript

JavaScript is a scripting language that can run wherever there is a suitable script interpreter such as Web browsers, Web servers, or the Windows Scripting Host.

Key Generator

This pertains to any tool designed to break software copy protection by extracting internally-stored keys, which can then be entered into the program to convince it that the user is an authorised purchaser.

Key

The Windows Registry uses keys to store computer configuration

settings. When a user installs a new program or the configuration settings are otherwise altered, the values of these keys change. If viruses modify these keys, they can produce damaging effects.

Logic Bomb

A logic bomb is a type of trojan horse that executes when specific conditions occur. Triggers for logic bombs can include a change in a file, by a particular series of keystrokes, or at a specific time or date. See also: Time Bomb

Macro

A macro is a series of instructions designed to simplify repetitive tasks within a program such as Microsoft Word, Excel or Access. Macros execute when a user opens the associated file. Microsoft's latest macro programming language is simple to use, powerful, and not limited to Word documents. Macros are mini-programs and can be infected by viruses. See also: Macro Virus

Macro Virus

A macro virus is a malicious macro. Macro viruses are written a macro programming language and attach to a document file (such as Word or Excel). When a document or template containing the macro virus is opened in the target application, the virus runs, does its damage and then copies itself into other documents. Continual use of the program results in the spread of the virus.

Mail Bomber

Software that floods a victim's inbox with hundreds or thousands of mails. Such mail generally does not correctly reveal its source.

Mailbomb

Excessively large e-mail (typically many thousands of messages) or many large messages sent to a user's e-mail account, for the purpose of crashing the system, or preventing genuine messages from being received.

Malware, Malicious Software

A generic term used to describe malicious software such as: viruses, trojan horses, malicious active content and others.

Malicious Code

A piece of code designed to damage a system or the data it contains, or to prevent the system from being used in its normal manner.

Master Boot Record, Master Boot Sector, MBR, MBS

The boot sector at the beginning of a hard drive (sector location 0,0,1 in CHS notation) is known as the master boot sector or, more commonly, the master boot record.

Master Boot Record Infector

A virus that infects master boot records.

Middle Infector

Refers to an entry point obscuring (EPO) virus. Due to design considerations in some scanners, some non-EPO viruses are referred to as middle infectors and may require special handling.

Multipartite Virus

A virus that infects two or more different target types is generally referred to as a multipartite virus. Early multipartite viruses infected boot sectors and DOS executables, but more esoteric combinations have been seen.

Mutex

MUTual EXclusion object. Mutex is a program object that allows multiple threads to share the same resource. Any thread that needs the resource must lock the mutex from other threads while it is using the resource.

Memory-Resident Virus

A memory-resident virus stays in memory after it executes and infects other files when certain conditions are met. Non-memory-resident viruses are active only while an infected application runs.

Mutating Virus

A mutating virus changes, or mutates, as it progresses through its host files making disinfection more difficult. The term usually refers to viruses that intentionally mutate, though some experts also include non-intentionally mutating viruses. See also: Polymorphic Virus

Network Creeper

Viruses that spread to new hosts by finding writable network drives (or 'shares') and copying themselves there or infecting files on those shares are sometimes referred to as network creepers.

Notifier

Any tool designed for stealth notification of an attacker that a victim has installed and run some pest. Such notification might be done by FTP, SMS, SMTP, or other method, and might contain a variety of information. Often used in combination with a Packer, a Binder and a Downloader.

Newsgroup

An electronic forum where readers post articles and follow-up messages on a specified topic. An Internet newsgroup allows people from around the globe discuss common interests. Each newsgroup name indicates the newsgroup's subject in terms of increasingly narrow categories, such as alt.comp.virus.

Oligomorphic Virus

An encrypted virus that has several forms of its decryption code, selecting between them (usually randomly) when writing its decryptor for a new replicant.

Overwriter

In general, the simplest form of virus is a program that just copies itself over the top of other programs. Such viruses are known as overwriters and are commonly the first types of viruses written for newly 'virused' platforms (e.g. Phage, the first PalmOS virus, discovered in late 2000, was a simple overwriter).

On-access Scanner

A real-time virus scanner that scans disks and files automatically, and often in the background. An on-access scanner scans files for viruses as the computer accesses the files.

On-demand Scanner

A virus scanner the user starts manually. Most on-demand scanners allow the user to set various configurations and to scan specific files, folders or disks.

On-schedule Scanner

A virus scanner the user schedules to start automatically at a given time.

Peer-to-Peer, P2P

Any peer-to-peer file swapping program, such as Audiogalaxy, Bearshare, Blubster, E-Mule, Gnucleus, Grokster, Imesh, KaZaa, KaZaa Lite, Limewire, Morpheus, Shareaza, WinMX and Xolox, in an organisation, can degrade network performance and consume vast amounts of storage. They may create security issues as outsiders are granted access to internal files. They are often bundled with Adware or Spyware.

Packer

A utility which compresses a file, encrypting it in the process. It adds a header that automatically expands the file in memory, when it is executed, and then transfers control to that file.

Parasitic Virus

Parasitic viruses are those that modify some existing code resource to effect replication.

Partition Boot Sector

The system boot sector of the active partition.

Partition Table

Partition tables are a crucial part of how DOS and related operating systems understand the layout of partitions (or logical

drives) on hard disks. For the sake of interoperability, most OSes that run on PCs also follow the dictates of these fundamental partition information resources.

Password Cracker

A tool to decrypt a password or password file-both for programs that take an algorithmic approach to cracking, as well as those that use brute force with a password cracking word list. Password crackers have legitimate uses by security administrators, who want to find weak passwords in order to change them and improve system security.

Password Cracking Word List

A list of words that a brute force password cracker can use to muscle its way into a system.

Payload

Refers to the effects produced by a virus attack. Sometimes refers to a virus associated with a dropper or Trojan horse.

Pervasiveness

Pervasiveness refers to a virus' potential to spread.

Polymorphic Virus

Polymorphic viruses create varied (though fully functional) copies of themselves as a way to avoid detection from anti-virus software. Some polymorphic virus use different encryption schemes and requires different decryption routines. Thus, the same virus may look completely different on different systems or even within different files. Other polymorphic viruses vary instruction sequences and use false commands in the attempt to thwart anti-virus software. One of the most advanced polymorphic viruses uses a mutation-engine and random-number generators to change the virus code and its decryption routine.

Port Scanner

In hacker reconnaissance, a port scan attempts to connect to all

65536 ports on a machine in order to see if anybody is listening on those ports.

POP3 or Post Office Protocol 3
A protocol that provides a simple, standardised way for users to access mailboxes and download messages to their computers.

Prepender
A virus that inserts a copy of its code at the beginning of the code of its victim file is known as a prepender or prepending virus.

Probe Tool
A tool that explores another system, looking for vulnerabilities. While these can be used by security managers, wishing to tighten up their security, the tools are as likely used by attackers to evaluate where to start an attack.

Proof of Concept, POC
It is commonly used to describe a virus that is the first to infect a given platform or implement a given infection technique.

Password Attacks
A password attack is an attempt to obtain or decrypt a legitimate user's password. Hackers can use password dictionaries, cracking programs, and password sniffers in password attacks. Defense against password attacks is rather limited, but usually consists of a password policy including a minimum length, unrecognisable words, and frequent changes.

Password Sniffing
The use of a sniffer to capture passwords as they cross a network. The network could be a local area network, or even the Internet itself. The sniffer can be hardware or software. Most sniffers are passive and only log passwords. The attacker must then analyse the logs later.

Piggyback
To gain access to a system via an authorised user's connection.

Program Infector

A program infector virus infects other program files once an infected application is executed and the activated virus is loaded into memory.

RAM, Random Access Memory

Memory transient programs are loaded into RAM so they can be executed. It is the memory that must be used for revisable data storage, regardless of the location of the program manipulating the data.

RAT, Remote Access Trojan, Remote Access Trapdoor

Remote Administration Tool. There are legitimate remote administration tools included with many network management products, with helpdesk and other support software, and the like. These are installed with the system administrator's knowledge and consent.

Registry

The registry is a database used by the Windows32 operating system (Win9x/ME/NT/2000/XP) to store configuration settings.

Remnant

There are many approaches to disinfecting virus-infected objects. As a result, some people are surprised to learn that not all products remove all traces of a virus when disinfecting. Should this happen, the remaining virus code will not be 'active'—it will not be able to gain control in the flow of execution—so the disinfected object is still 'safe'. These snippets of leftover code are sometimes referred to as remnants.

Resident Virus

A resident virus loads into memory and remains inactive until a trigger event. When the event occurs the virus activates, either infecting a file or disk, or causing other consequences. All boot viruses are resident viruses and so are the most common file viruses.

Retro-virus

Loosely based on the biological concept with the same name, com-

puter viruses that attack antivirus products are sometimes referred to as retro-viruses.

Real-time Scanner
An anti-virus software application that operates as a background task, allowing the computer to continue working at normal speed, with no perceptible slowing.

Redirect
The action used by some viruses to point a command to a different location. Often this different location is the address of the virus and not the original file or application.

Rename
The action by which a user or program assigns a new name to a file. Viruses may rename program files and take the name of the file so running the program inadvertently runs the virus.

Repliction
The process by which a virus makes copies of itself in order to carry out subsequent infections. Replication is one of major criteria separating viruses from other computer programs.

Resident Extension
A resident extension is a memory-resident portion of a program that remains active after the program ends. It essentially becomes an extension to the operating system. Many viruses install themselves as resident extensions.

ROM, Read-Only Memory
Apart from its contents normally not being modifiable, ROM is usually also non-volatile. This type of memory is traditionally used to hold a PC's BIOS and little else, although various kinds of 'modifiable ROM' memory technologies, such as EPROM, EEPROM and flash memory, have been used through the years, with flash memory being preferred in recent years.

Rogue Program

A term the media use to denote any program intended to damage programs or data, or to breach a system's security. It includes trojan horse programs, logic bombs, viruses, and more.

Search Hijacker

Any software that resets your browser's settings to point to other sites when you perform a search.

Self-Encrypting Virus

Self-encrypting viruses attempt to conceal themselves from anti-virus programs. Most anti-virus programs attempt to find viruses by looking for certain patterns of code (known as virus signatures) that are unique to each virus. Self-encrypting viruses encrypt these text strings differently with each infection to avoid detection. See Self-garbling Virus, Encrypted Virus

Self-Garbling Virus

A self-garbling virus attempts to hide from anti-virus software by garbling its own code. When these viruses spread, they change the way their code is encoded so anti-virus software cannot find them. A small portion of the virus code decodes the garbled code when activated. See Also: Self-encrypting Virus, Polymorphic Virus.

Signature

A search pattern, often a simple string of characters or bytes, expected to be found in every instance of a particular virus. Usually, different viruses have different signatures. Anti-virus scanners use signatures to locate specific viruses. Also: Virus Signatures

Slow Mailer

A slow mailer is a virus that distributes itself from victim machines via e-mail but not in the 'explosive' manner attributed to mass mailers.

Slow Polymorphism

A term occasionally applied to polymorphic viruses that only morph their code 'occasionally' rather than each time they replicate, as is more common. This is an 'anti-antivirus research' technique.

SMTP

Simple Mail Transport Protocol. The Internet e-mail delivery format for transmitting e-mail messages between servers.

Sniffer

A wiretap that eavesdrops on computer networks. The attacker must be between the sender and the receiver in order to sniff traffic. This is easy in corporations using shared media. Sniffers are frequently used as part of automated programs to sift information off the wire, such as clear-text passwords, and sometimes password hashes (to be cracked).

Social Engineering

There are two main ways to obtain technical or administrative information about a computer system. The first is from the systems themselves and the second is from the administrators and users of the machines. Surreptitious or unauthorised attempts to obtain such system information are known as hacking or cracking if the attempt involves obtaining information from the machines, and is called social engineering if the attempts involve manipulating or 'tricking' a person into divulging the information.

SOCKS Proxy

Socks (or SOCKS) is an IETF standard protocol for TCP/IP-based networking applications. A proxy server (a server that sits between a client application and a real server) can use SOCKS to accept requests from clients so that they can be forwarded across the Internet. Socks uses sockets to represent and keep track of individual connections.

SPAM Tool

Any software designed to extract e-mail addresses from Web sites

and other sources, remove 'dangerous' or 'illegal' addresses, and/or efficiently send unsolicited (and perhaps untraceable) mail to these addresses.

Sparse Infector
Just like slow infection methods, sparse infection is also an approach to reduce the chances of early detection. The main idea is to replicate only occasionally; for example, only infecting one in every 100 programs that are executed.

Spyware
A program that gathers information and can be 'silently' installed and run in 'stealth' mode. This kind of software is used to gather information from a user's machine, such as recorded keystrokes (passwords), a list of Web sites visited by the user, applications installed on the machine, the version of operating system, registry settings and so on.

Spyware Cookie
Any cookie that is shared among two or more unrelated sites for the purpose of gathering and sharing private user information.

Stealth Virus
Stealth viruses attempt to conceal their presence from anti-virus software. Many stealth viruses intercept disk-access requests, so when an anti-virus application tries to read files or boot sectors to find the virus, the virus feeds the program a 'clean' image of the requested item. Other viruses hide the actual size of an infected file and display the size of the file before infection.

Surveillance
Any software designed to use a Web cam, microphone, screen capture, or other approaches to monitor and capture information. Some such software will transmit this captured information to a remote source.

SYN Flood Attack

In the normal course of a TCP connection, a SYN (TCP connection request) is sent to a target computer. When the target computer receives the SYN, it sends a SYN_RECEIVED message back to the machine that sent the SYN (reading the IP source address of the originating packet). The target computer then waits for the machine that originated the request to send back a SYN_ACK upon receipt of its SYN_RECEIVED message (this SYN-RECEIVED state is saved in a buffer either until the ACK is received or until the request has been waiting for a particular finite period of time and is then purged). When this three-way handshake is completed, data can travel freely between the two computers.

Telnet Server

Software that allows a remote user of a Telnet client to connect as a remote terminal from anywhere on the Internet and control a computer in which the server software is running.

Time Bomb

A logic bomb with its trigger condition(s) based on absolute or elapsed date or time conditions.

TOM

Top of Memory. A design limit at the 640 KB-mark on most PCs. Often the boot record does not completely reach top of memory, thus leaving empty space. Boot sector infectors often try to conceal themselves by hiding around the top of memory. Checking the top of memory value for changes can help detect a virus, though there is also non-viral reasons this value change.

Tracking Cookie

Any cookie that is shared among two or more Web pages for the purpose of tracking a user's surfing history.

Trigger

The condition that determines the launching of a virus' or Trojan's payload is usually called the trigger or trigger condi-

tion. Trigger is also used as a verb to indicate the activation of a payload.

Trojan, Trojan Horse
A Trojan horse program is a malicious program that pretends to be a benign application; a Trojan horse program does something the user does not expect. Trojans are not viruses since they do not replicate, but they can be just as destructive.

Trojan Creation Tool
A program designed to create Trojans. Some of these tools merely wrap existing Trojans, to make them harder to detect. Others add a trojan to an existing product (such as RegEdit.exe), making it a Dropper.

Trojan Source
Source code is written by a programmer in a high-level language and readable by people but not computers. Source code must be converted to object code or machine language before a computer can read or execute the program. Trojan Source can be compiled to create working trojans, or modified and compiled by programmers to make new working trojans.

Timestamp
The time of creation or last modification recorded on a file or another object. Users can usually find the timestamp in the Properties section of a file.

TSR
Terminate and Stay Resident. TSR programs stay in memory after being executed. TSR programs allow the user to quickly switch back and forth between programs in a non-multitasking environment, such as MS-DOS. Some viruses are TSR programs that stay in memory to infect other files and programs. Also: Memory-resident Program

Tunnelling
A virus technique designed to prevent anti-virus applications

from working correctly. Anti-virus programs work by intercepting the operating system actions before it can execute a virus. Tunnelling viruses try to intercept actions before the anti-virus software can detect the malicious code. New anti-virus programs can recognise many viruses with tunnelling behaviour.

Usage Tracks

Usage tracks permit any user (or their software agent) with access to your computer to see what you've been doing. Such tracks benefit you if you have left the tracks, but might benefit another user as well.

Virus

A computer program file capable of attaching to disks or other files and replicating itself repeatedly, typically without user knowledge or permission. Some viruses attach to files so when the infected file executes, the virus also executes. Other viruses sit in a computer's memory and infect files as the computer opens, modifies or creates the files. Some viruses display symptoms, and some viruses damage files and computer systems, but neither symptoms nor damage is essential in the definition of a virus; a non-damaging virus is still a virus.

Virus Creation Tool

A program designed to generate viruses. Even early virus creation tools were able to generate hundreds or thousands of different, functioning viruses, which were initially undetectable by current scanners.

Web Bug

A Web Bug is a device used in HTML Web pages and e-mail that is used to monitor who is reading the Web page or e-mail. These are small, hidden, difficult to detect eavesdropping devices. Most of the time, you will not even be aware that these bugs exist, as they hide within 1 by 1 pixel html image tags, although any graphic on a Web page or in an e-mail can be configured to act as a Web bug. This is not to say that all invisible gifs on Web pages are Web bugs; some invisible gif files are used for alignment and design purposes.

WildList

Although there are many thousands of known viruses, few actually cause any real-world concern, and those that do are often said to be 'in the wild'. However, the term 'in the wild' has been used in many different contexts and with many different shades of meaning. In an attempt to clear this situation up, for computer viruses, antivirus researcher Joe Wells instigated what he called the WildList. Its purpose was to provide a listing of viruses that could (or should) be considered 'in the wild' by set criteria.

Worm

Worms are parasitic computer programs that replicate, but unlike viruses, do not infect other computer program files. Worms can create copies on the same computer, or can send the copies to other computers via a network.

Worm Creation Tool

A program designed to generate worms. Worm creation tools can often generate hundreds or thousands of different, functioning worms, most of which are initially undetectable by current scanners.

Windows Scripting

Windows Scripting Host (WSH) is a Microsoft integrated module that lets programmers use any scripting language to automate operations throughout the Windows desktop.

Zoo

A collection of viruses used for testing by researchers.

Tools

Knowledge is power. When it comes to viruses and security threats, the more you know, the more prepared you can be. We have covered just about all the important subjects related to viruses that an average computer user should know, but you can always learn more. Luckily, the information is easy to find. We have listed a selection of useful books and Web sites that can help you learn just about everything you need to know about defending yourself against various threats arising from malicious software or otherwise.

Computer Viruses For Dummies

Author: Peter H Gregory
Publisher: John Wiley & Sons, Inc

Just the thought of your trusty PC catching a computer virus is probably enough to make you sick. Thanks to the annoying virus writers who persist in coming up with new strains, there's a major new cyberattack nearly every day. Viruses just happen to sneak in from various sources. Fortunately, there are ways to inoculate and protect your computer.

Computer Viruses For Dummies helps you understand the risks and analyse your PC's current condition. It also helps in selecting, installing, and configuring the antivirus software along with giving you information on scanning your computer and e-mail and ridding your computer of viruses that it's already caught.

There's helpful information on the use of firewalls and spyware blockers, protecting handheld PDAs from viruses and adopting safe computing practices, especially with e-mail and when you are surfing the Internet.
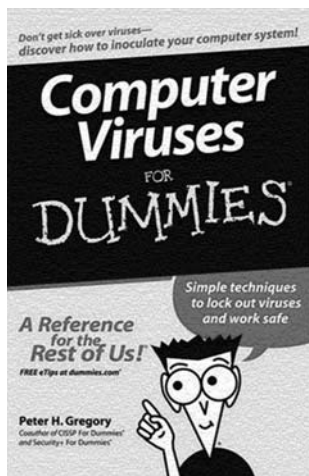
A Short Course on Computer Viruses

Author: Frederick Cohen
Publisher: John Wiley & Sons, Inc

Here is an outstanding opportunity to learn about computer viruses from the internationally acclaimed pioneer in the field who actually coined the phrase "computer virus." This new edition of Cohen's classic work has been updated and expanded to nearly double its original size and now includes entirely new chapters on LAN viruses, international viruses, and good viruses (including code).
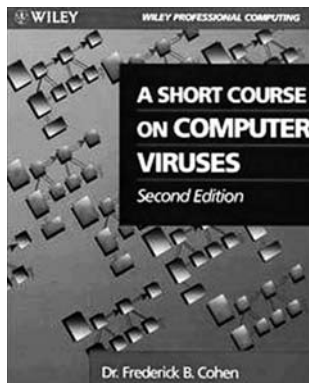
As entertaining as it is thorough, the text is enlivened by Cohen's down-to-earth wit and his many fascinating anecdotes and as yet unpublished historical facts about viruses. Both broad in its coverage and deep in its consideration, it includes dozens of lucid explanations and examples that amicably guide the reader through the complex, often convoluted subject matter. Hailed as a tour de force, Cohen's discussion of defensive strategies reveals many of the stumbling blocks that often trip readers up.

Virus Proof, Second Edition

Author: Phil Schmauder
Publisher: Prima Publishing

Like biological viruses, computer virus-
es can spread quickly and are often dif-
ficult to get rid of without causing dam-
age. *Virus Proof: The Ultimate Guide to
Protecting Your System* provides key steps
you should take to protect your system
from these destructive viruses. Inside
you will learn how to recover data that
is lost as a result of a virus, what com-
mon viruses do, and how they spread.

Virus Proof is an excellent resource for any computer user, from
beginners to experts.
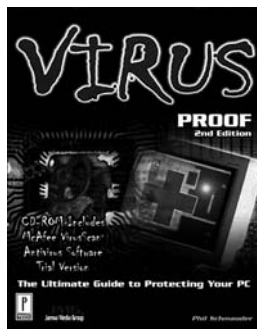
Blocking Spam For Business For Dummies

Authors: Peter H Gregory, Mike Simon
Publisher: John Wiley & Sons, Inc

Despite recent legislation in the
US and other countries, the vol-
ume of spam continues to grow.
Spam now accounts for 60 to 75
per cent of all e-mail.

*Blocking Spam For Business For
Dummies* shows small business
people and corporate information
security professionals how to fight
back successfully against this
onslaught, offering savvy advice
on selecting and deploying a spam
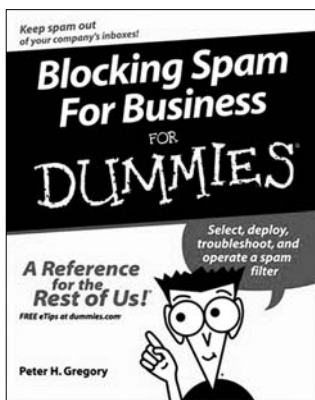
filter as well as training and supporting users. It also provides insider tips on troubleshooting and fine-tuning a spam filter, as well as exclusive guidance on how to deal with 'Joe Jobs' spam attacks, in which spammers hijack a corporate domain name.

Malware: Fighting Malicious Code

Authors: Ed Skoudis, Lenny Zeltser
Publisher: Prentice Hall PTR

Ignoring the threat of malware is one of the most reckless things you can do in today's increasingly hostile computing environment. Malware is malicious code planted on your computer, and it can give the attacker a truly alarming degree of control over your system, network, and data-all without your knowledge! Written for computer pros and savvy home users by computer security expert Edward Skoudis, *Malware: Fighting Malicious Code* covers everything you need to know about malware, and how to defeat it!
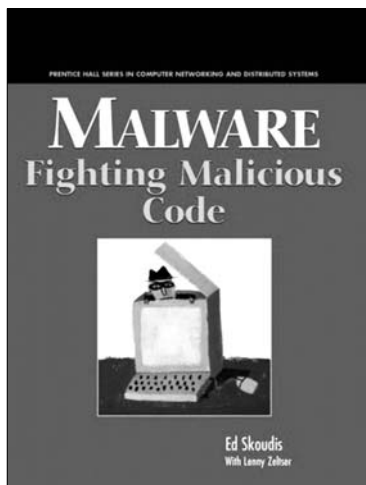
This book devotes a full chapter to each type of malware—viruses, worms, malicious code delivered through Web browsers and e-mail clients, backdoors, Trojan horses, user-level RootKits, and kernel-level manipulation. You'll learn about the characteristics and methods of attack, evolutionary trends, and how to defend against each type of attack. Real-world examples of malware attacks help

you translate thought into action, and a special defender's toolbox chapter shows how to build your own inexpensive code analysis lab to investigate new malware specimens on your own. Throughout, Skoudis' clear, engaging style makes the material approachable and enjoyable to learn.

This book includes solutions and examples that cover both UNIX and Windows operating systems. There are practical, time-tested, real-world actions you can take to secure your systems specified in this book, along with instructions for building your own inexpensive malware code analysis lab so you can get familiar with attack and defensive tools harmlessly!

E-mail Virus Protection Handbook: Protect Your E-mail From Viruses, Trojan Horses, And Mobile Code Attacks

Authors: Brian Bagnall, James Stanger
Publisher: Syngress Publishing

E-mail has been called the killer application of the Internet with so many Web-based commerce applications, business-to-business transactions, and Application Service Providers dependent on the e-mail client/server relationship. Now, because of that reliance, it is possible for e-mail software to become killer applications in an entirely different sense—if they're down, they can kill your business. E-mail Virus Protections Handbook will help systems administrators and end-users
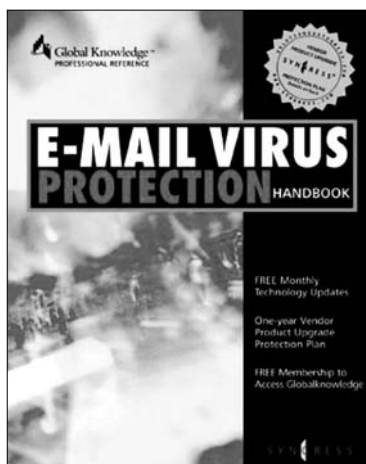
secure their e-mail. It shows how to encrypt e-mail messages, use antivirus and personal firewall software, and secure the operating system from attack. Know what's lurking in your e-mail!

Topics covered include malicious code that's spread through e-mail clients, servers, and protocols, and how to defend against it. Specifically, the book deals with antivirus software-both network-wide and for single clients-and configuration policies for Outlook 2000, Outlook Express 5.0, and Eudora 4.3 on the client side. Server coverage includes Windows 2000 Advanced Server, Red Hat Linux 6.0, Exchange Server 5.5, and Sendmail. Personal firewalls, such as BlackICE Defender 2.1, get attention, too.

How To Do Everything To Fight Spam, Viruses, Pop-Ups, And Spyware

Author: Ken Feinstein
Publisher: McGraw-Hill Osborne Media

Get expert advice on finally ridding your computer of annoying spam and pop-up ads and invasive viruses, spyware, and adware. You will discover where these electronic nuisances originate, how they work, and how to pre-vent them. Learn to choose spam-resistant e-mail address-es and get the most from your spam filter. Protect your com-puter from virus attacks with antivirus software and pre-ventive measures. Also, find out how to avoid installing spyware and adware unknow-ingly, and block those pesky pop-up ads. The bonus CD-ROM features trial versions of the prevention tools covered in the book.

The book helps you understand how spammers operate and how to safeguard your e-mail addresses along with using spam-filtering software and challenge-response mail systems. It shows you how to configure your PC to resist virus attacks and recognise virus-laden e-mails and avoid virus infection when downloading files. There's vital information on installing, configuring, and updating antivirus software, and diagnosing and removing viruses from your system; and also tips on how to avoid installing adware and spyware inadvertently.

Counter Hack: A Step-by-Step Guide To Computer Attacks And Effective Defenses

Author: Ed Skoudis
Publisher: Prentice Hall PTR

This next-generation hacker book gives you a step-by-step guide to defending against hacker intrusions. Articles feature how to defend against today's most powerful hacker attacks; detect intrusion using new evasion techniques and countermeasures. It's written by Edward Skoudis, the security expert who demonstrated hacking to the US Senate.
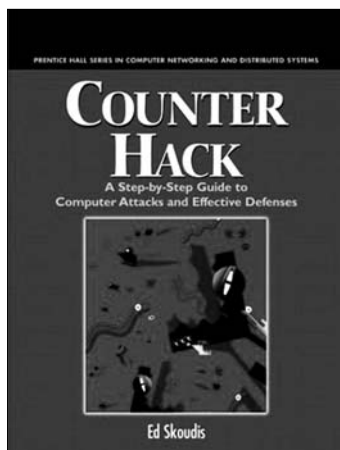
This easy-to-use, step-by-step guide will empower network and system administrators to defend their information and computing assets-whether or not they have security experience. In Counter Hack, leading network security expert Edward Skoudis presents comprehensive, insider's explanations of today's most destructive hacker tools and tactics-and

specific, proven countermeasures for both UNIX and Windows environments.

Skoudis covers all this and more in topics such as:

❍ Know your adversary: from script kiddies to elite attackers.
❍ A hacker's view of networks, TCP/IP protocols, and their vulner-abilities.
❍ Five phases of hacking: reconnaissance, scanning, gaining access, maintaining access, and preventing detection.
❍ The most dangerous and widespread attack scenarios-explained in depth.
❍ Key hacker tools: port scanners, firewall scanners, sniffers, session hijackers, RootKits, and more.
❍ How hackers build elegant attacks from simple building blocks
❍ Detecting and preventing IP spoofing, covert channels, denial of service attacks, and other key attacks.
❍ How hackers cover their tracks-and how you can uncover their handiwork.
❍ A preview of tomorrow's hacker tools, attacks, and counter-measures.

Whatever your role in protecting network infrastructure and data, *Counter Hack* delivers proven solutions you can implement right now-and long-term strategies that will improve security for years to come.

Firewalls For Dummies, Second Edition

Authors: Brian Komar, Ronald Beekelaar, Joern Wettern
Publisher: John Wiley & Sons, Inc

What an amazing world we live in! Almost anything you can imagine can be researched, compared, admired, studied, and in many cases, bought, with the click of a mouse. The Internet has changed our lives,

Find out how to build barriers that keep cyber-invaders at bay

Firewalls
FOR
DUMMIES

2nd Edition

Discover the best ways to protect your privacy online

A Reference for the Rest of Us!
FREE eTips at dummies.com®

Brian Komar
Ronald Beekelaar
Joern Wettern, PhD
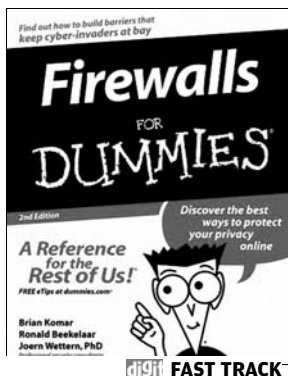
putting a world of opportunity before us. Unfortunately, it has also put a world of opportunity into the hands of those whose motives are less than honourable. A firewall, a piece of software or hardware that erects a barrier between your computer and those who might like to invade it, is one solution.

If you've been using the Internet for any length of time, you've probably received some unsavoury and unsolicited e-mail. If you run a business, you may be worried about the security of your data and your customers' privacy. At home, you want to protect your personal information from identity thieves and other shady characters. *Firewalls For Dummies* will give you the low-down on firewalls, then guide you through choosing, installing, and configuring one for your personal or business network.

Firewalls For Dummies helps you understand what firewalls are, how they operate on different types of networks, what they can and can't do, and how to pick a good one. You will find out about developing security policies, establishing rules for simple protocols, detecting and responding to system intrusions, setting up firewalls for SoHo or personal use, creating demilitarised zones, using Windows or Linux as a firewall, configuring ZoneAlarm, BlackICE, and Norton personal firewalls and installing and using ISA server and FireWall-1.

With the handy tips and hints this book provides, you will find that firewalls are nothing to fear-unless you're a cyber-crook! You will soon be able to keep your data safer, protect your family's privacy, and probably sleep better, too.

Absolute Beginner's Guide To Personal Firewalls

Authors: Jerry Ford, Stephen Dodd
Publisher: Que

A consumer-level firewall guide committed to teaching how to

choose the right firewall software, set up a personal firewall, and test computer security. Personal firewall security is particularly useful for the ever-increasing number of users with 'always on' Internet connections such as those with a cable modem or DSL connection.

While previous firewall books have been focused on network professionals and network firewall protection.

These books have not adequately addressed the consumer's need for personal firewall protection. This book is designed to provide simplified, yet thorough firewall information on the most prevalent personal firewall software applications available for the non expert firewall consumer.

This book will walk readers through the basics such as determining the need for a firewall and testing current security. Other chapters will demonstrate and explain: how to tighten security, choose a high-speed Internet connection, install a personal firewall, and test new security.
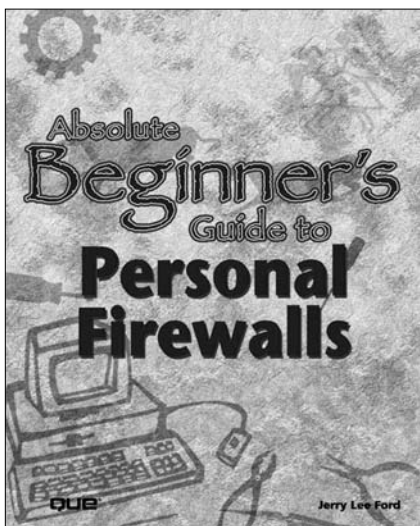
Firewalls: The Complete Reference

Authors: Keith Strassberg, Gary Rollie, Richard Gondek
Publisher: McGraw-Hill Osborne Media

Get in-depth, objective advice on installing and configuring today's most popular firewalls including Check PointTM Firewall-

1 4.1 and NG, Cisco PIX, Microsoft ISA Server, NetScreen, SonicWall and Symantec-and learn strategies for successful network design and firewall placement. Gain insight into common methods for attacking firewalls-including software bugs, viruses, and misconfigurations. Learn firewall best practices and how to improve the overall security of your firewall installation. This multipurpose guide contains all the implementation and administration information you need to keep your network safe from unauthorised access.
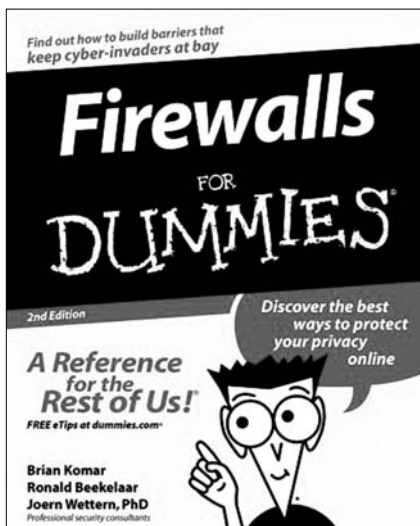
Learn to restrict access to your network without compromising usability and functionality. Understand the strengths and limitations of firewall technology with the in-depth explanations of network and port address translation, VPNs, authentication, virus protection, content filtering, and more. Learn about the various architectures available today-application and circuit-level gateways, packet filters, and stateful packet inspection engines.

Find out how hackers commonly go about breaking into a network. Manage your firewall installation using inbuilt tools, objects, and services and supplement it by implementing human controls such as education and log monitoring.

The Art of Computer Virus Research And Defense

Author: Peter Szor
Publisher: Addison-Wesley Professional

Symantec's chief antivirus researcher has written the definitive guide to contemporary virus threats, defence techniques, and analysis tools. Unlike most books on computer viruses, *The Art of Computer Virus Research and Defense* is a reference written strictly for white hats: IT and security professionals responsible for protecting their organizations against malware. Peter Szor systematically covers everything you need to know, including virus behavior and classification, protection strategies, antivirus and worm-blocking techniques, and much more.

Szor presents the state-of-the-art in both malware and protection, providing the full technical detail that professionals need to handle increasingly complex attacks. Along the way, he provides extensive information on code metamorphism and other emerging techniques, so you can anticipate and prepare for future threats.

Szor also offers the most thorough and practical primer on virus analysis ever published-addressing everything from creating your own personal laboratory to automating the analysis process. This book's coverage includes discovering how malicious code attacks on a variety of platforms, classifying malware strategies for infection, in-memory operation, self-protection, payload delivery, exploitation, and more. It also includes identifying and responding to code obfuscation threats: encrypted, polymorphic, and metamorphic, mastering empirical methods for analyzing malicious code-and what to do

with what you learn, reverse-engineering malicious code with disassemblers, debuggers, emulators, and virtual machines.

The book teaches you how to implement technical defences such as scanning, code emulation, disinfection, inoculation, integrity checking, behaviour blocking, using worm blocking, host-based intrusion prevention, and network-level defence e-strategies.

Web sites

Virus Bulletin

www.virusbtn.com
Virus Bulletin started in 1989 as a magazine dedicated to providing PC users with a regular source of intelligence about computer viruses, their prevention, detection and removal, and how to recover programs and data following an attack.

Editorial independence has always been Virus Bulletin's prime concern. From the very first issue, Virus Bulletin has cut through antivirus hype and remained uninfluenced by sales pitches and marketing babble. The aim of the magazine is to arm users with all the information they need to stay current with the latest developments in the antivirus field.

Symantec Worldwide Home Page

www.symantec.com

The premier site for most people who get hit by a new worm! Symantec is not merely just a home to one of the world's most
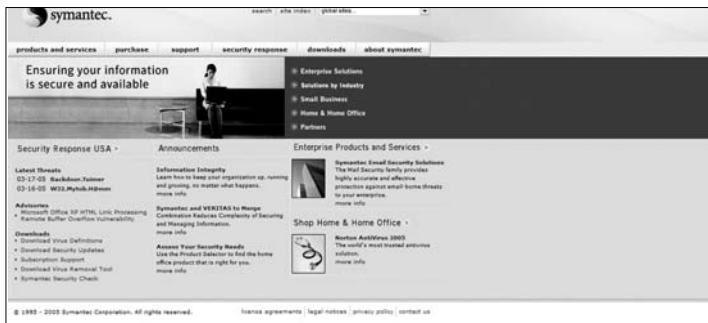
widely used antivirus solution, but it's considered a Mecca for people who want to follow up on the latest developments in the antivirus industry.

Vmyths

www.**vmyt**hs.com

The term 'Computer Virus' can be quite scary for some people, and for those who don't have the knowledge about it can be led to believe anything. That's where Vmyths comes in.
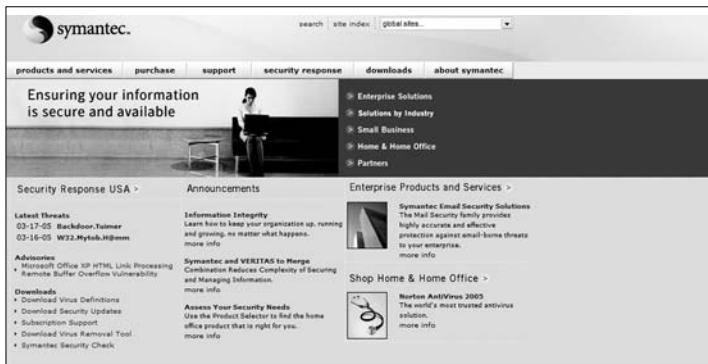
Vmyths helps you learn about computer virus myths, hoaxes, urban legends, hysteria, and the implications if you believe in them. You can also search a list of computer virus hoaxes and virus hysteria to broaden your knowledge about the truth behind the scare. Vmyths stays independent from any antivirus companies and claims to speak of the truth rather than what the antivirus companies want you to believe.

Microsoft Security

www.microsoft.com/security
For the latest information in security fixes, updates and patches for Windows and other Microsoft products, this site is the place to be. Along with security updates, Microsoft security provides news and information on recent developments in the field; and also contain guides on keeping your PC more secure.