PEARSON IT CYBERSECURITY CURRICULUM

FOURTH EDITION

# COMPUTER SECURITY FUNDAMENTALS

DR. CHUCK EASTTOM

# Computer Security Fundamentals

**Fourth Edition**

Dr. Chuck Easttom

Pearson

### The Mimail Virus

Mimail is another older virus that is still worth studying. The Mimail virus did not receive as much media attention as Sobig, but it had some intriguing characteristics. This virus collected email addresses not only from your address book but also from other documents on your machine. Thus, if you had a Word document on your hard drive and an email address was in that document, Mimail would find it. This strategy meant that Mimail would spread further than many other viruses. Mimail had its own built-in email engine, so it did not have to "piggyback" off your email client. It could spread regardless of what email software you used.

These two differences from most viruses make Mimail interesting to people who study computer viruses. There are a variety of techniques that allow you to programmatically open and process files on your computer; however, most virus attacks do not employ them. The scanning of the document for email addresses indicates a certain level of skill and creativity on the part of the virus writer. In this author's opinion, Mimail was not the work of an amateur but rather a person with professional-level programming skill.

### The Bagle Virus

Bagle is another historical virus. It is not an issue today but is noteworthy in the history of malware because it combined email attachments with a fake virus warning. The Bagle virus began to spread rapidly in the fourth quarter of 2003. The email it sent claimed to be from your system administrator. It would tell you that your email account had been infected by a virus and that you should open the attached file to get instructions. Once you opened the attached file, your system was infected. This virus was particularly interesting for several reasons. To begin with, it spread both through email and by copying itself to shared folders. Second, it could scan files on your PC, looking for email addresses. Finally, it would disable processes used by antivirus scanners. In biological terms, this virus took out your computer's "immune system." The disabling of virus scanners was a new twist that indicated at least moderate programming skills on the part of the virus creator.

### A Nonvirus Virus

Another new type of virus has been gaining popularity in the past few years, and that is the "nonvirus virus" or, put simply, a hoax. Rather than actually writing a virus, a hacker sends an email to every address he has. The email claims to be from some well-known antivirus center and warns of a new virus that is circulating. The email instructs people to delete some file from their computer to get rid of the virus. The file, however, is not really a virus but part of a computer's system. The jdbgmgr.exe virus hoax used this scheme. It encouraged the reader to delete a file that was actually needed by the system. Surprisingly, a number of people followed this advice and not only deleted the file but promptly emailed their friends and colleagues to warn them to delete the file from their machines.

### FYI: The Morris Internet Worm

The Morris worm was one of the first computer worms ever to be distributed over the Internet—and it was certainly the first to gain any significant media attention.

Robert Tappan Morris, Jr., then a student at Cornell University, wrote this worm and launched it from an MIT system on November 2, 1988. Morris did not actually intend to cause damage with the worm. Instead, he wanted the worm to reveal bugs in the programs it exploited in order to spread. However, bugs in the code allowed an individual computer to be infected multiple times, and the worm became a menace. Each additional "infection" spawned a new process on the infected computer. At a certain point, the large number of processes running on an infected machine slowed down the computer to the point of being unusable. At least 6000 UNIX machines were infected with this worm.

Morris was convicted of violating the 1986 Computer Fraud and Abuse Act and was sentenced to a $10,000 fine, 3 years' probation, and 400 hours of community service. But perhaps the greatest impact of this worm was that it led to the creation of the Computer Emergency Response Team (CERT). CERT (www.cert.org) is an organization hosted at Carnegie Mellon University that is a repository for security bulletins, information, and guidelines. CERT is a source that any security professional should be familiar with.

### Flame

No modern discussion of viruses would be complete without a discussion of Flame. This virus, which first appeared in 2012, targeted Windows operating systems. The first item that makes this virus notable is that it was specifically designed by the U.S. government for espionage. It was discovered in May 2012 at several locations, including Iranian government sites. Flame is spyware that can monitor network traffic and take screenshots of the infected system.

### The Earliest Viruses

It is instructive to consider the very first viruses every found. In 1971, Bob Thomas created what is widely believed to be the first computer virus, named Creeper. It spread through the ARPANET (the precursor to the Internet) and displayed a message "I'm the creeper, catch me if you can!" Another program, named Reaper, was created to delete Creeper.

Wabbit, which was found in 1974, made multiple copies of itself, thus adversely affecting the performance of the infected computer.

Apple Viruses 1, 2, and 3 are some of the first viruses "in the wild" or public domain. These viruses, which were found on the Apple II operating system in 1981, spread through Texas A&M via pirated computer games.

### The Impact of Viruses

In early 2018, Taiwan Semiconductor Manufacturing Company, one of the largest chipmakers and a supplier for Apple, said it had been hit by a computer virus that had affected computer systems and fabrication tools. Estimates placed the damages over $170 million. The specific virus was not described in the news reports, but a single company being hit with a single virus causing so much havoc illustrates the dangers of computer viruses.

### Rules for Avoiding Viruses

You should notice a common theme with all virus attacks (except the hoax): They want you to open some type of attachment. The most common way for a virus to spread is as an email attachment. Knowing this, you can follow a few simple rules to drastically reduce the odds of becoming infected with a virus:

- Use a virus scanner. McAfee and Norton (explored in the exercises at the end of this chapter) are the two most widely accepted and used virus scanners. However, Kaspersky and AVG are also good, reputable choices. Each costs about $30 per year to keep your virus scanner updated. Do it. Each antivirus product has proponents and detractors, and I won't delve into the opinions on which is better. For most users, any of the four major antivirus programs would be effective. I rotate which one I use periodically just so I can stay familiar with all of them.
- If you are not sure about an attachment, do not open it.
- You might even exchange a code word with friends and colleagues. Tell them that if they wish to send you an attachment, they should put the code word in the title of the message. Without seeing the code word, you will not open any attachment.
- Do not believe "security alerts" that are sent to you. Microsoft does not send out alerts in this manner. Check the Microsoft website regularly, as well as one of the antivirus websites previously mentioned.

These rules will not make your system 100% virus proof, but they will go a long way toward protecting your system.

## Trojan Horses

Recall from earlier chapters that *Trojan horse* is a term for a program that looks benign but actually has a malicious purpose. We have already seen viruses that are delivered via a Trojan horse. You might receive or download a program that appears to be a harmless business utility or game. More likely, the Trojan horse is just a script attached to a benign-looking email. When you run the program or open the attachment, it does something else other than or in addition to what you thought it would. It might do any of the following:

- Download harmful software from a website.
- Install a key logger or other spyware on your machine.
- Delete files.
- Open a backdoor for a hacker to use.

It is common to find combination virus/Trojan horse attacks. In those scenarios, the Trojan horse spreads like a virus. The MyDoom virus opened a port on your machine that a later virus, doomjuice, would exploit, thus making MyDoom a combination of a virus and a Trojan horse.

A Trojan horse can also be crafted especially for an individual. If a hacker wished to spy on a certain individual, such as the company accountant, he could craft a program specifically to attract that person's attention. For example, if he knew the accountant was an avid golfer, he could write a program that computed handicap and listed best golf courses. He would post that program on a free web server. He would then email a number of people, including the accountant, telling them about the free software. The software, once installed, could check the name of the currently logged-on person. If the logged-on name matched the accountant's name, the software could then go out, unknown to the user, and download a key logger or other monitoring application. If the software did not damage files or replicate itself, then it would probably go undetected for quite a long time. There have been a number of Trojan horses through the years. One of the earliest and most widely known was Back Orifice.

### FYI: Virus or Worm?

As noted in Chapter 4, there is disagreement among the experts about the distinction between a virus and a worm. Some experts would call MyDoom (as well as Sasser, which will be discussed later) a worm because it spread without human intervention. However, I would define a virus as any file that can self-replicate and a worm as any program that can propagate without human interference. This is also the most common definition you will find among security experts.

Such a program could be within the skill set of virtually any moderately competent programmer. This is one reason that many organizations have rules against downloading *any* software onto company machines. I am unaware of any actual incident of a Trojan horse being custom tailored in this fashion. However, it is important to remember that those creating virus attacks tend to be innovative people.

It is also important to note that creating a Trojan horse does not require programming skill. There are free tools on the Internet, such as EliteWrapper, that allow someone to combine two programs—one hidden and one not. So one could easily take a virus and combine it with, for example, a poker game. The end user would see only the poker game, but when it was run, it would launch the virus.

Another scenario to consider is one that would be quite devastating. Without divulging programming details, the basic premise will be outlined here to illustrate the grave dangers of Trojan horses. Imagine a small application that displays a series of unflattering pictures of Osama Bin Laden. This application would probably be popular with many people in the United States, particularly people in the military, intelligence community, or defense-related industries. Now assume that this application simply sits dormant on the machine for a period of time. It need not replicate like a virus because the computer user will probably send it to many of his associates. On a certain date and time, the software connects to any drive it can, including network drives, and begins deleting all files. If such a Trojan horse were released "in the wild," within 30 days it would probably be shipped to thousands, perhaps millions, of people. Those thousands or millions of computers would then begin deleting files and folders. Imagine the devastation.

This scenario is mentioned precisely to frighten you a little. Computer users, including professionals who should know better, routinely download all sorts of things from the Internet, such as amusing flash videos and cute games. Every time an employee downloads something of this nature, there is a chance of downloading a Trojan horse. One need not be a statistician to realize that if employees continue that practice long enough, they will eventually download a Trojan horse onto a company machine. If they do, hopefully the virus will not be as vicious as the theoretical one just outlined here.

Because Trojan horses are usually installed by users themselves, the security countermeasure for this attack is to prevent downloads and installations by end users. From a law enforcement perspective, the investigation of a crime involving a Trojan horse would involve a forensic scan of the computer hard drive, looking for the Trojan horse itself.

There are a number of tools, some free for download, that will help a person create a Trojan horse. One that I use in my penetration testing classes is eLiTeWrap. It is easy to use. Essentially, it can bind any two programs together. Using a tool such as this one, anyone can bind a virus or spyware to an innocuous program such as a shareware poker game. This would lead to a large number of people downloading what they believe is a free game and unknowingly installing malware on their own system.

The eLiTeWrap tool is a command line tool, but it is very easy to use. Just follow these steps:

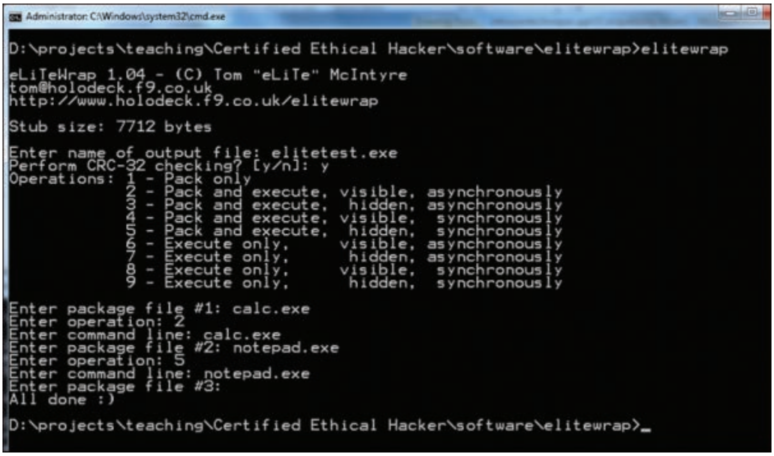
1. Enter the file you want to run that is visible.
2. Enter the operation:
   - 1—Pack only
   - 2—Pack and execute, visible, asynchronously
   - 3—Pack and execute, hidden, asynchronously
   - 4—Pack and execute, visible, synchronously
   - 5—Pack and execute, hidden, synchronously
   - 6—Execute only, visible, asynchronously

   - 7—Execute only, hidden, asynchronously
   - 8—Execute only, visible, synchronously
   - 9—Execute only, hidden, synchronously

3. Enter the command line.
4. Enter the second file (the item you are surreptitiously installing).
5. Enter the operation.
6. When done with files, press Enter.

In Figure 5.1 you can see a demonstration that is appropriate for a classroom laboratory. In this example, two innocuous programs are combined into one Trojan horse. The programs chosen are simple Windows utilities that won't harm the computer. However, it illustrates how easy it would be to combine legitimate programs with malware, for delivery to a target computer.

This illustration is meant to show how easy it is to create a Trojan horse, not to encourage you to do so. It is important to understand just how easy this process is so you can understand the prevalence of malware. Any attachment or download should be treated with significant suspicion.



```
D:\projects\teaching\Certified Ethical Hacker\software\elitewrap>elitewrap

eLiTeWrap 1.04 - (C) Tom "eLiTe" McIntyre
tom@holodeck.f9.co.uk
http://www.holodeck.f9.co.uk/elitewrap

Stub size: 7712 bytes

Enter name of output file: elitetest.exe
Perform CRC-32 checking? [y/n]: y
Operations: 1 - Pack only
            2 - Pack and execute, visible, asynchronously
            3 - Pack and execute,  hidden, asynchronously
            4 - Pack and execute, visible,  synchronously
            5 - Pack and execute,  hidden,  synchronously
            6 - Execute only,     visible, asynchronously
            7 - Execute only,      hidden, asynchronously
            8 - Execute only,     visible,  synchronously
            9 - Execute only,      hidden,  synchronously

Enter package file #1: calc.exe
Enter operation: 2
Enter command line: calc.exe
Enter package file #2: notepad.exe
Enter operation: 5
Enter command line: notepad.exe
Enter package file #3:
All done :)

D:\projects\teaching\Certified Ethical Hacker\software\elitewrap>_
```

**FIGURE 5.1** eLiTeWrap.

## The Buffer-Overflow Attack

You have become knowledgeable about a number of ways to attack a target system: denial of service, virus, and Trojan horse. While these attacks are probably the most common, they are not the only methods. Another method of attacking a system is called a buffer-overflow (or buffer-overrun) attack.