Computer Worms: Past, Present, and Future

Craig Fosnock
CISSP, MCSE, CNE
East Carolina University

Abstract:

Internet computer worms have gone from a hypothetical theorem to very real and very dangerous threat to computer networks. They are even capable of affecting the biggest network of our time, the Internet. Starting from humble and beneficial beginnings computer worms are now the plague of the Internet and can cause billions of dollars worth of damages in just a few hours, if not minutes. In this paper I will discuss the history of computer worms, their past, present, and their possible future, but before we start this discussion about computer worms lets first define some computer terminology. It is important that we have these definitions up front not only because we will need them to help explain the rest of this paper, it is also important to discuss this now because the term computer virus is often used interchangeably to refer to computer worms. This may cause confusion with some readers as this paper is focusing on computer worms, not computer viruses. I have chosen not to address viruses because, although viruses take advantage of network services such as the Internet to spread, viruses are currently somewhat less common than worms, and viruses do not seem to have emulated the scale of disruptive behavior, and monetary damage that current computer worms are capable of inflecting on today's computer networks.

Computer Program:
Is an example of computer software that prescribes the actions ("computations") that are to be carried out by a computer. Most programs consist of a loadable set of instructions which determines how the computer will react to user input when that program is running, i.e., when the instructions are 'loaded'. The term program or computer program is used interchangeably with software and software application. ("Computer Program," 2005)

Computer Virus:
Is a self-replicating program that spreads by inserting copies of itself into other executable code or documents. A computer virus behaves in a way similar to a biological virus, which spreads by inserting itself into living cells. Extending the analogy, the insertion of the virus into a program is termed infection, and the infected file (or executable code that is not part of a file) is called a host. ("Computer Virus," 2005)

Computer Worm:
Is a self-replicating computer program, similar to a computer virus but unlike a virus which attaches itself to, and becomes part of, another executable program, a worm is self-contained and does not need to be part of another program to propagate itself. In other words a typical computer virus is similar to a parasite and it requires a host. In this case the host is another executable program. A worm does not need a host, it can spread on its own. ("Computer Worm," 2005)

I. The Past

A. Background

The roots of the modern computer virus go back to John Von Neumann. Von Neumann's name because of his publication of the concept, was given to the von Neumann architecture. This architecture is used in most non-parallel-processing computers. Almost every commercially available home computer, microcomputer and supercomputer is a von Neumann machine. In 1949 von Neumann created the field of cellular automata only using pencil and graph paper, when he presented a paper on the "Theory and Organization of Complicated Automata." In this paper he postulated that a computer program could reproduce. The paper included a model of what is now known as a computer virus.

In the 1950s Bell Labs employees gave life to von Neumann's theory in a game they called "Core Wars." This game was created by H. Douglas McIlroy, Victor Vysottsky, and Robert Morris, Sr. The object of the game was to unleash software "organisms" they called "self-altering automata" that attacked, erased, and tried to propagate in a computer generated world. Each organism was a small program consisting of well-defined instructions with each of the instructions occupying a single cell in a memory array. In modern times this computer-generated world is actually a linear looping array of memory cells called the "Core," but in the 50's the word core referred to magnetic core memory. The game was started when two programs were loaded into random positions in the core. After a certain number of cycles, if neither program has quit executing, a tie was declared. If one program terminated, the other was declared the winner. The primary objective behind the game was to write an organism that could terminate all the opponent processes. Re-energized and popularized by A.K. Dewdney through his series of articles in Scientific American, Core Wars and its variants can still be found and played on the Internet or on your local computer. The "organisms" now called "warriors"that are used in the Core Wars is considered the forebear of the modern computer virus including computer worms.

B. The First worms

The first computer worm was created at the legendary Xerox PARC (Palo Alto Research Center). For those of you who do not know this research center not only created the first computer worm it gave us the first personal computer, the first graphical user interface and the first laser printer. The worm was created by John Shoch. Shoch was a PARC engineer working on his Stanford doctorate when he created the worm. The program took its name from the "tapeworm," a program that appeared in a popular science-fiction novel of the time by John Brunner called "The Shockwave Rider." This science-fiction novel ironically helped to promote the concept of a replicating program more than other more serious writings on the subject.

Unlike the worm in the book Shockwave Rider which was used to destroy a sinister computer network, the PARC worm's intended purpose was save Shoch hours of tedious work. Shoch's doctoral research was an analysis of the traffic patterns of PARC's Ethernet (another PARC first) that linked 200 of its "Altos," personal computers. His idea was to arrange for about 100 of the machines to spew bits into the Ethernet simultaneously, then measure the ensuing electronic gridlock. ("Benefits," n.d) Rather than loading the same program individually into every machine, he devised the worm to do the loading automatically by seeking out idle Altos computers and transmitting the test program by wire to those that signaled they were available. The test proved successful and soon he turned his thoughts from communicating directly with each machine to instructing them to talk among themselves. Shoch eventually was able to invest his worm with the ability to seek out idle Altos, boot up a host machine through the network and replicate by sending copies of itself from machine to machine, remaining in communication with its dispersed offspring.

One night, however, something unexpectedly went wrong. Shoch and two colleagues had set a small worm loose in the PARC Ethernet to test a control function, and went home. At some point the program became corrupted so badly it crashed its host computer. Sensing it had lost a segment, the control worm sent out a tendril to another idle Alto. That host crashed, and the next, and the next. For hours, the worm spread through the building until scores of machines were disabled. The next day the down machines did not cause any alarm as Altos frequently crashed for no reason. Soon, however, it became obvious that this was no random occurrence. Summoned to the lab Shoch and his colleagues could not stop the worm and eventually they had no choice but to eradicate the worm with a failsafe software mechanism that Shoch had pre loaded as insurance against some unpredictable disaster. ("Benefits," n.d)

The next worm started out as a joke or innocent prank, but is among the first and most notable worms to qualify as a network exploit. The worm was launched from Germany on December 9, 1987. This date is well before the Internet was officially born. The worm originated on the German EARN network, propagated through connected Bitnet sites and eventually worked its way through Bitnet connections to wreak havoc on the IBM Internal File Transfer Network (otherwise known as VNET) in the United States. (Henry, 2003) The name given to this worm was the Christma Exec, which happened to be the name of the script the user needed to execute to launch the worm.

As mentioned above Christma Exec required the user to execute an innocent looking script that was attached to an E-mail message, which appeared to come from an E-mail address that the recipient knew and trusted. This ruse sounds really familiar, and as we should all know by now that this ruse is still an effective means of computer worm distribution. When the user executing the script it would cause a Christmas tree to appear on the terminal and then it mailed itself to everyone on the user's NAMES file including any distribution lists. When it finished sending itself to all addressees, it erased itself from the original victim's computer. When new recipients received and activated their copies of Christma Exec, the scenario would repeat, flooding the network with Christma Exec messages. The flood of traffic created left parts of the IBM network unusable on December 10 and 11 until it was finally brought under control. (Henry, 2003)

Even with the introduction of two fully functional worms, most people still treated computer worms as an obscure theoretical problem. That perception of worms took a dramatic turn in late-1988, when a college student, and son of the above mentioned and co-creator of the “Core Wars” Robert Morris, Sr. unleashed the infamous "Internet Worm," otherwise known as the Morris worm or the “Great worm,” on the new and unsuspecting Internet.

The Morris worm was a “Multi Mode” worm that attacked DEC VAX servers running Sun and BSD operating systems. It exploited weak passwords along with known vulnerabilities in the send mail application and Unix utilities fingerd and rsh/rexec. Although not, written to cause damage a bug in Robert’s software allowed the worm to reinfect individual servers multiple times. Hence, each additional instance of the worm on the server caused additional CPU resources to be consumed, slowing the server and effectively causing the world’s first Internet denial of service (DoS) attack. (Henry, 2003) At the time of the attack it was estimated that the Morris worm infected approximately 6,000 servers or 10% of the servers on the Internet, and caused between $10 million and $100 million in damages.

II. The Present

The development of computer worms seemed to die down until the development of the Melissa worm eleven (11) years later. Here is a time line and brief synopsis of modern computer worms. You will notice that the time between virus outbreaks, and the estimated amount of damages will be increasing. You will also notice no entries for the year 2002. This is because Klez which dominated that year was released in 2001, and none of the worms created that year although destructive like the BugBear worm, did not provided any new computer worm developments.

Year 1999

Virus name: Melissa
Description: First found in March 26, 1999, using holes in Microsoft Outlook, Melissa shut down Internet mail systems that got clogged with infected e-mails propagating from the worm. Once executed the original version of Melissa used a macro virus to spread to the first 50 addresses in the user's Outlook address book. However, if Internet access or Outlook were not available, it would copy itself to other word documents and attempt to E-mail those documents, revealing potentially confidential information. Further, it would modify existing documents by inserting quotes from the Simpson's television show. (Henry, 2003)
Estimated damage: $1.1 billion.

Year 2000

Virus name: I LOVE YOU
Description: First found on May, 3, 2000 in Asia it spread quickly across the globe. Instead of sending a copy of the worm to the first 50 or 100 addresses in the host's Outlook address book like Melissa, I Love You used every single address in the host's address book. This worm also had a malicious side to it, as the worm overwrote important files with a copy of itself, making it virtually impossible to recover original files. It also marked all mp3 files as hidden, and downloaded a Trojan horse that would steal user names and passwords and them to the virus's author.
Estimated damage: $8.75 billion.

Year 2001

Virus name": Anna Kournikova Virus" worm
Description: First appearing in February 2001 it was produced by a "scrip kiddie," and is well known only for its social engineering attachment that appeared to be a graphic image of Russian tennis star Anna Kournikova. However, when the file was opened, a clandestine code extension enabled the worm to copy itself to the Windows directory and then send the file as an attachment to all addresses listed in your Microsoft Outlook e-mail address book. The "Anna Kournikova Virus" worm although famous was just a nuisance as it did little to no damage
Estimated damage: $166,827

Virus name: Code Red
Description: First found on July 13, 2001 this worm exploited a vulnerability in Microsoft's Internet Information Server (IIS) web servers to deface the host's website, and copy the command.com file and rename it root.exe in the Web server's publically accessible scripts directory. This would provide complete command line control to anyone who knew the Web server had been compromised. It also waited 20-27 days after it was installed to launch denial of service attacks against the White House's IP address. Code Red spread at a speed that overwhelmed network administrators as more than 359,000 servers became compromised in just over 14 hours. At its peak, more than 2,000 servers were being compromised every single minute. Estimates are that Code Red compromised more than 750,000 servers. (Henry, 2003)
Estimated damage: $2.6 billion

Virus name: Sircam

Description: First found on July 19, 2001 this mass mailing E-mail worm not only exploited Microsoft's Outlook program it had the ability of spreading through Windows Network shares. The worm had two deadly payloads, but due to a program error they did not work.
Estimated damage: $1.03 billion

Virus name: NIMDA
Description: First appearing in September 2001, NIMDA, which is admin spelled backwards was not as malicious in nature as previous worms, but its advanced features and its different means of propagation which included from client to client via email, from client to client via open network shares, from web server to client via browsing of compromised web sites, from client to web server via active scanning for and exploitation of various Microsoft IIS vulnerabilities, and from client to web server via scanning for the back doors left behind by the "Code Red II" and "sadmind/IIS" worms, allowed it to spread faster than any preceding worm. NIMDA also the first worm that contained its own E-mail program so it did not depend on the host's E-mail program to propagate.
Estimated damage:$645 million

Virus name: Klez
Description: First appearing in October 26, 2001 Klez, and it variants were still considered a problem late in 2003, making Klez one of the most persistent viruses ever. Klez was a hybrid worm that took advantage of a flaw in Outlook that allowed it to be installed simply by viewing the E-mail in the preview panel. As a hybrid threat it could behave like a virus, a worm and at other times even like a Trojan horse. Klez also incorporated a technique we saw in the Christma Exec worm as it selected one E-mail address from the host's address book to use as the "from" address, then sending the worm to all the other addresses. In this manner, the E-mail often appeared to have been sent from someone the addressee actually knew.
Estimated damage: $18.9 billion

Year 2003.

Virus name: SQL Slammer
Description: Appearing January 25, 2003, and taking advantage of two buffer overflow bugs in Microsoft's SQL Server database product, it spread rapidly, with a doubling time of 8.5 seconds in the early phases of the attack allowing it to infecting most of its victims within 10 minutes. SQL Slammer was the first example of a "Warhol worm." A Warhol worm was first hypothesized in 2002 in a paper by Nicholas Weaver, and it is an extremely rapidly propagating computer worm that spreads as fast as physically possible, infecting all vulnerable machines on the entire Internet in 15 minutes or less. The term is based on Andy Warhol's remark that "In the future, everybody will have 15 minutes of fame." (Computer Worm, 2005)
Estimated damage: $1.2 billion.

Virus name: Sobig
Description: Originally put together in January 2003 to spread a proxy server trojan, its variant Sobig.F set a record in sheer volume of e-mails. Sobig like Nimda used a built-in SMTP engine so it did not depend on the host's E-mail program to propagate. Then emulating Klez, it selected one E-mail address from the host's address book to use as the "from" address, then sending the worm to all the other addresses. It also attempted to create a copy of itself on network shares, but failed due to bugs in the code.
Estimated damage: $36.1 billion

Virus name: Blaster
Description: Appearing August 11, 2003 Blaster exploited a Microsoft DCOM RPC vulnerability to infect systems running Windows 2000 and Windows XP, and cause instability on systems running Windows NT, and Windows Server 2003. Filtering of virus activity by Internet service providers (ISPs) worldwide greatly reduced the spread of Blaster.
Estimated damage: $1.3 billion

Year 2004

Virus name: Mydoom
Description: Appearing January 26, 2004 and primarily transmitted via E-mail to appear as a transmission error, Mydoom's rapid spread becomes the fastest spreading email worm ever. It slowed overall Internet performance by about 10%, and average web page load times by about 50%.
Estimated damage: $38.5 billion

Virus name: Witty
Description: Appearing March 19, 2004, the Witty worm was the fastest developed worm to date as there was only 36 hours between the release of the advisory to the release of the virus. Witty infected the entire exposed population of twelve thousand machines in 45 minutes, and it was the first widespread worm that destroyed the hosts it infected (by randomly erasing a section of the hard drive) without significantly slowing the worm's expansion.
Estimated damage: $11 million

Virus name: Sasser
Description: Appearing on April 30, 2004 and spreading by exploiting a buffer overflow in the component known as LSASS, (Local Security Authority Subsystem Service) it hit the Internet a little more than two weeks after Microsoft warned users of this flaw. Although it caused infected Windows XP and Windows 2000 computers to repeatedly reboot, Sasser did little damage, as was merely designed to spread and carried no payload.
Estimated damage: $14.8 billion

Although all the computer viruses discussed above in the time line are malicious in nature, not all computer worms are meant to be bad. These viruses are often called "beneficial viruses" or "antivirus" viruses because they attack other viruses and disinfect them from the systems that they have compromised. An early example of this is the Den_Zuko boot virus37, which was actually a worm that disinfected the Brain virus. The Brain virus was a malicious code created in Pakistan which infected boot sectors of disks so that their contents could not be accessed. Brain was the first PC virus created and it infected MS-DOS. Another more resent example of this behavior is found in the Nachi family of worms, which terminated and deleted the Blaster worm, then tried to download and install patches to fix the Microsoft DCOM RPC vulnerability in the host system. Although considered "beneficial" in nature both of these worms cause problems. In the case of Nachi it generated more network traffic than the Blaster worm it was protecting against. In the case of Den_Zuko boot virus37 it could not infect 1.2M or 3.5" diskettes correctly, and because of this it destroyed data on them. Most importantly both worms worked without the explicit consent of the computer's owner or user. Because of these problems no true "beneficial" worm has been created, whatever their payload.

II. The Future

It is expected that in the future that we will see ever more complex worms labeled "Super Worms." These worms will incorporating complex polymorphic, and metamorphic behavior routines that will make use of entry-point obscuration. The current trends in traditional areas of worm development do not seem to be leading us immediately in the direction of the "Super Worm," but they are getting their slowly. SpyBot.KEG is an example the new batch of worms attempting to use these behavior methods. It is considered a sophisticated vulnerability assessment worm, and it set new standards for all computer worms. SpyBot.KEG has managed to remain below most people's radar as it causes no damage, and only reports discovered vulnerabilities back to the author via IRC channels. Security experts have also seen the release of multiple variants per a single day of another sophisticated worm called Mytob. This worm has recently included a phishing trick in the form of a fake URL pointing to a Web site that hosts the worm's code.

Another possible future threat that could develop into the next “Super Worm” outbreak is called YellowFever. YellowFever is an advanced i-worm with some really interesting features, which shows that conceptual complexity of current i-worms in the wild is well far from what can be done. A Short virus description: the worm installs itself as a system service. On startup, it enumerates all running applications looking for its target (Outlook). The infection procedure is very interesting: the virus has a small built-in debugger that uses to attach to the host. Next, it impersonates the host and, using its own "SMTP" engine, E-mails itself. "YellowFever" is not polymorphic but it would be possible to add a poly-engine to it. The virus can bypass many of the user level firewalls, but it has not been coded for spreading. (Labir, 2005) Although complex these worms have failed to be modified to the point where they can cause any major damage, but that could change at anytime.

Other new worm developments include the Cabir worm. The Cabir worm is the first worm that can infect mobile phones. Cabir appears to be a so-called "proof of concept" worm and requires social engineering to reach its goal but once a phone is infected, it will activate each time the phone is started, scan to nearby Bluetooth enabled phones, and transmit a copy of itself to the any vulnerable phone it reaches.

The most prevalent and immediately dangerous worms that seem to be developing are those designed to propagate via instant messaging (IM). Although older but less well known worms like the Hello worm used Microsoft's MAN Messenger to spread, the current batch of worms using IM seems to hold the most promise of being the next major outbreak. When we take a closer look at this theat we can see that there are about 60 published IM vulnerabilities, and the types of IM threats are expanding to include SPIM (spam over IM) and phishing attacks. As we have already seen, propagation speed for worms is limited only by their ability to find new hosts. In the case of Code Red took about 14 hours to ping every IP address in the world looking for vulnerable systems, and in Slammer’s case it took only 20 minutes. A similar threat targeting IM, according to a Symantec simulation could lead to half a million systems being infected in only 30 seconds. This is because the worm would already have a list of vulnerable machines located on users’ “ buddy” lists. This any new worm using IM would not have to use time-consuming methods to locate vulnerable systems. Once inside a vulnerable company, an IM worm would cause a lot of damage as it would bypass all existing security defenses.

Could the next major worm outbreak not even involve computers? Will the outbreak be a new development in traditional propagation methods or will it come from new areas such as IM? Regardless history has shown us that no matter what method it will use to propagate, it is only a matter of time before we see the next major worm outbreak.

References

Benefits of the Computer Virus. (Unknown)
Owl Editing
Retrieved July 15, 2005, from
http://www.owled.com/essays/virus.html

Blaster Worm. (2005).
Wikipedia, the free encyclopedia.
Retrieved July 15, 2005, from
http://en.wikipedia.org/wiki/Blaster_worm

Cabir. (2005).
Wikipedia, the free encyclopedia.
Retrieved July 15, 2005, from
http://en.wikipedia.org/wiki/Cabir

Cathleen Moore (2005)
Worms wiggle into IM. *Computer World*Retrieved July 15, 2005, from
http://www.computerworld.com /securitytopics/ security/virus/story/0,10801,101201,00. html?source=x10

Computer Worm. (2005).
Wikipedia, the free encyclopedia.
Retrieved July 15, 2005, from
http://en.wikipedia.org/wiki/Computer_worm

Computer program. (2005).
Wikipedia, the free encyclopedia.
Retrieved July 15, 2005, from
http://en.wikipedia.org/wiki/Computer_program

Computer virus . (2005).
Wikipedia, the free encyclopedia.
Retrieved July 15, 2005, from
http://en.wikipedia.org/wiki/Computer_virus

Chen, T.M.. (2003). Trends in viruses and worms. *The Internet Protocol Journal, Vol. 6,* No. 3.

Daniel Tynan (2003)
Dawn of the Superworm. *PC World*.
Retrieved July 15, 2005, from
http://www.pcworld.com/news/article /
0,aid,110014,tfg,tfg,00.asp

Edward Skoudis. (2002, July) Malware - The Worm Turns. *Information Security Magazine*.

Eduardo Labir (2004) VX reversing II, Sasser B. *The CodeBreakers-Journal*, Vol. 1, No. 1

George V. Hulme (2005)
First Worm For Mobile Phones Detected.
Information Week .
Retrieved July 15, 2005, from
http://www.informationweek.com/story/showArticle.jhtml?articleID=21800340

Gregg Keizer (2005)
IM Worms Could Spread In Seconds. *Security Pipeline*
Retrieved July 15, 2005, from
http://www.securitypipeline.com/news/22100839

Ingrid Marson. (2005)
Mytob worm picks up phishing trick. *News* .
Retrieved July 15, 2005, from
http://news.com.com/Mytob+worm+picks+up+phishing+trick/2100-7349_3-5739271.html?part=rss&tag=5739271&subj=news

ILOVEYOU. (2005).
Wikipedia, the free encyclopedia.
Retrieved July 15, 2005, from
http://en.wikipedia.org/wiki/VBS/Loveletter

Jack M. Germain (2004)
Is the Superworm a Mere Myth?.
Tech News World.
Retrieved July 15, 2005, from
http://www.technewsworld.com/story/32721.html

Julia Allen. (2002) System and Network Security Practices. *Journal of Information Security,* Vol1, No 2

Klez. (2005).
Wikipedia, the free encyclopedia.
Retrieved July 15, 2005, from
http://en.wikipedia.org/wiki/Klez

Mary Landesman. (2005)
Mytob prevention. *About*.
Retrieved July 15, 2005, from
http://antivirus.about.com/od/
virusdescriptions/a/mytob.htm

Melissa worm (2005).
Wikipedia, the free encyclopedia.
Retrieved July 15, 2005, from
http://en.wikipedia.org/wiki/Melissa_worm

Morris worm (2005).
Wikipedia, the free encyclopedia.
Retrieved July 15, 2005, from
http://en.wikipedia.org/wiki/Morris_worm

Opic (2004) Introductory Primer To Polymorphism. *The CodeBreakers-Journal*, Vol. 1, No. 2

Paul Henry. (2003).
A Brief Look at the Evolution of Killer Worms.
CyberGuard.
Retrieved July 15, 2005, from
http://www.csoonline.com/whitepapers /
050504_cyberguard
/EvolutionoftheKillerWorms.pdf

Peter Szor. (2005)
Strategies of Computer Worms. *The Informit Network*
Retrieved July 15, 2005, from
http://www.informit.com/articles/article.asp?p=366891&seqNum=8&rl=1

Peter Westrin. (2001) Critical Information Infrastructure Protection (CIIP). *Information & Security: An International Journal*. Volume 7, 67-79

Rinku Dewr (2003)
Artificial Life: A Programmer's Perspective.
Retrieved July 15, 2005, from
http://ai-depot.com/ArtificialLife/
Programmer-Perspective.html

Scarlet Pruitt (2005)
Are Virus Writers Creating a Super Worm?. *PC World*
Retrieved July 15, 2005, from

Sasser worm.(2005).
Wikipedia, the free encyclopedia.
Retrieved July 15, 2005, from
http://en.wikipedia.org/wiki/Sasser worm

Symantec, *Symantec Antivirus Software and Information.* 2005.