The Evolution of Viruses and Worms

Thomas M. Chen
Dept. of Electrical Engineering
SMU
PO Box 750338
Dallas, TX 75275-0338 USA
Tel: 214-768-8541
Fax: 214-768-3573
Email: tchen@engr.smu.edu

Jean-Marc Robert
Alcatel Canada Inc.
600 March Road
Ottawa, Ontario, Canada K2K 2E6
Tel: (613) 784-5988
Email: jean-marc.robert@alcatel.com

Abstract - Computer viruses and network worms have evolved through a continuous series of innovations, leading to the recent wave of fast-spreading and dangerous worms. A review of their historical development and recent outbreaks leads to a number of observations. First, while viruses were more common than worms initially, worms have become the predominant threat in recent years, coinciding with the growth of computer networking. Second, despite widespread use of firewalls and other network security equipment, worm outbreaks still occur and will likely continue to be a threat for the near future. Third, recent worms are appearing as a series of quick successive variants. Unlike the independent efforts of early viruses, these variants suggest an increasing level of coordination among worm creators. Fourth, recent worms have shown capabilities to spread faster and exploit more infection vectors. This trend implies a more urgent need for automated, coordinated protection measures. Finally, more dangerous payloads are becoming commonplace. This suggests that worm creators are using worms for other objectives than simply infection, such as data theft and setting up denial of service networks.

1. Introduction

Computer viruses and worms are characterized by their ability to self replicate. The modern computer virus was conceived and formalized by Fred Cohen as a USC graduate student in 1983. Cohen wrote and demonstrated the first documented virus in November 1983 [1]. Like biological viruses, computer viruses reproduce by taking advantage of the existing environment. A biological virus consists of single or double-stranded nucleic acid (DNA or RNA) surrounded by a protein shell (capsid). The capsid gives specificity to bond with those particular hosts with matching surface receptors, while the inner nucleic acid gives infectivity or potency to subvert the infected host's cellular machinery. A virus is incomplete and inactive outside of a living cell but becomes active within a host cell by taking over the host's metabolic machinery to create new virus particles that spread the infection to other cells.

Computer viruses replicate themselves by attaching their program instructions to an ordinary "host" program or document, such that the virus instructions are executed during the execution of the host program. As a

simplification of Cohen's definition, a basic computer virus can be viewed as a set of instructions containing at least two subroutines attached somehow to or within a host program or file [2]. The first subroutine of the virus carries out the infection by seeking out other programs and attaching or overwriting a copy of the virus instructions to those programs or files [3]. The method of infection or propagation is called the "infection vector" (borrowed from epidemiology). The second subroutine of the virus carries the "payload" that dictate the actions to be executed on the infected host. The payload could be almost anything in theory, for example, deletion of data, installation of backdoors or DoS (denial of service) agents, or attacks on antivirus software [4]. An optional third subroutine could be a "trigger" that decides when to deliver the payload [5].

Computer networks have created a fertile environment for worms, which are related to viruses in their ability to self-replicate but are not attached parasitically to other programs. Worms are stand-alone automated programs designed to exploit the network to seek out vulnerable computers to infect with a copy of themselves. In contrast to viruses, worms are inherently dependent on a network and not dependent on any human action (such as to execute a program infected with a virus). Worms have become more prevalent since Internet connectivity has become ubiquitous. The Internet increases the vulnerability of all interconnected computers by making it easier for malicious programs computers to move among computers.

Recent worm outbreaks, such as the Blaster worm in August 2003 and the SQL Sapphire/Slammer worm in January 2003, have demonstrated that networked computers continue to be vulnerable to new attacks despite the widespread deployment of antivirus software, firewalls, intrusion detection systems, and other network security equipment. We review the historical development of viruses and worms to show how they have evolved in sophistication over the years. The history of virus and worm evolution is classified into four “waves” spanning from the Shoch and Hupp worms in 1979 to the present day (the term “generations” is less preferred because viruses and worms of one wave are not direct descendents from an earlier wave). This historical perspective leads to a number of observations about the current state of vulnerability and trends for possible future worm attacks.

We classify the evolution of viruses and worms into the following waves:

• first wave from 1979 to early 1990s

• second wave from early 1990s to 1998

• third wave from 1999 to 2001

• fourth wave from 2001 to today.

These waves represent periods where new technological trends began and appeared in a significant number of cases. For example, the third wave was dominated by so-called mass e-mailers: viruses and worms that exploited e-mail programs to spread. The classification dates are approximate and not meant to imply that waves can be separated so distinctly. For example, mass e-mailers characteristic of the third wave are still common during the fourth wave.

2. First Wave: Early Viruses and Worms

The first wave of viruses and worms from roughly 1979 to early 1990s were clearly experimental. The early viruses were commonly boot-sector viruses and targeted mostly to MS DOS. The early worms were prone to programming bugs and typically hard to control. The term “worm” was created by John Shoch and Jon Hupp at Xerox PARC in 1979, inspired by the network-based multi-segmented "tapeworm" monster in John Brunner's novel, *The Shockwave Rider*. Shoch and Hupp used worm to refer to any multi-segmented computation spread over multiple computers. They were inspired by an earlier self-replicating program, Creeper written by Bob Thomas at BBN in 1971, which propelled itself between nodes of the ARPANET. However, the idea of self-replicating programs can be traced back as early as 1949 when the mathematician John von Neumann envisioned specialized computers or “self-replicating automata” that could build copies of themselves and pass on their programming to their progeny [6].

Shoch and Hupp invented worms to traverse their internal Ethernet LAN seeking idle processors (after normal working hours) for distributed computing [7]. Since the worms were intended for beneficial uses among cooperative users, there was no attempt at stealth, intrusion, or malicious payload. Even under cooperative conditions however, they observed that worm management was a key problem to ensure that worm growth could be reliably contained. Their worms were designed with limited lifetimes, and responsive to a special packet to kill all worms. Despite these safeguards, one of the worm programs mysteriously ran out of control and crashed several computers overnight.

In 1983, Fred Cohen conceived, wrote and demonstrated the first computer virus while a graduate student at USC. The 1986 DOS-based Brain virus, supposedly written by two Pakistani programmers, was interesting in its

attempt to stealthily hide its presence by simulating all of the DOS system calls that normally detect viruses, causing them to return information that gave the appearance that the virus was not there.

In 1987, the "Christma Exec" virus was among the first to spread by e-mail among IBM mainframes. It is also an early example of social engineering where the user is tricked into executing the virus because it promised to draw a Christmas tree graphic. The worm does produce a Christmas card graphic on the computer screen (drawn using a scripting language called Rexx) but also sends a copy of itself in the user's name to his list of outgoing mail recipients. The recipients believe the e-mail is from the user so they open the e-mail.

On November 2, 1988, the famous Morris worm disabled 6,000 computers in a few hours (constituting 10 percent of the Internet at that time) [8]. The worm was written by a Cornell student, Robert Morris Jr. Investigations conducted later (ending in his conviction in 1990) concluded that his motives for writing the worm were unknown but the worm was not programmed deliberately for destruction. Instead, the damage appeared to be caused by a combination of accident and programming bugs. It was detected and contained only because an apparent programming error caused it to re-infect computers that were already infected, resulting in a noticeable slowdown in infected computers. It was among the first to use a combination of attacks to spread quickly: cracking password files; exploiting the debug option in the Unix “sendmail” program; and carrying out a buffer overflow attack through a vulnerability in the Unix “finger” daemon program.

In October 1989, the WANK (Worms Against Nuclear Killers) worm apparently learned from the Morris worm and infected VMS computers on DECnet. It spread using e-mail functions, exploited default system and field service accounts and passwords for access, and tried to find accounts where the user name and password were the same or the password was null.

3. Second Wave: Polymorphism and Toolkits

The second wave in the approximate period from the early 1990s to 1998 saw much more activity in viruses than worms, although the technical advances in viruses would effect the later evolution of worms. In this period, viruses began to move from Microsoft DOS to Windows as the primary target; cross-platform macro viruses appeared; mass creation of polymorphic viruses became easy; and a trend towards e-mail as the preferred infection vector began.

In the late 1980s, the idea of using encryption to scramble the appearance of a virus was motivated by the fact that antivirus software could detect viruses by scanning files for unique virus signatures (byte patterns). However, to be executable, an encrypted virus must be prepended with a decryption routine and encryption key. The decryption routine remains unchanged and therefore detectable, although the key can change which scrambles the virus body differently. Polymorphism carries the idea further to continuously permute the virus body. A polymorphic virus reportedly appeared in Europe in 1989. This virus replicated by inserting a pseudorandom number of extra bytes into a decryption algorithm that in turn decrypts the virus body. As a result, there is no common sequence of more than a few bytes between two successive infections.

Polymorphism became a practical problem in 1992 when a well-known hacker, Dark Avenger, developed a user-friendly Mutation Engine to provide any virus with polymorphism [9]. Other hackers soon followed with their own versions of mutation engines with names such as TPE, NED, and DAME. In 1994, Pathogen and Queeg were notable polymorphic DOS-infecting viruses that were produced by Black Baron's SMEG (Simulated Metamorphic Encryption enGine) [5].

The “virus creation lab” was a user-friendly, menu-driven programming toolkit that allowed hackers with little programming skill to easily generate hundreds of new viruses [9]. Other virus creation toolkits soon followed such as PS-MPC. Perhaps the best known product of a virus toolkit was the 2001 Anna Kournikova virus [10]. This virus was carried in an e-mail attachment pretending to be a JPG picture of the tennis player. If the Visual Basic Script attachment is executed, the virus e-mails a copy of itself to all addresses in the Outlook address book.

The 1995 Concept virus was the first macro virus, written for Word for Windows 95. It infected Word's "normal.dot" template so that files were saved as templates and ran the infective AutoOpen macro. An infected file was accidentally shipped on a Microsoft CD called "Microsoft Windows95 Software Compatibility Test." Later, Microsoft UK shipped an infected file on another CD called "The Microsoft Office 95 and Windows95 Business Guide." The vast majority of macro viruses are targeted to Microsoft Office documents. Macro viruses have the advantages of being easy to write and cross-platform. However, most people now know to disable macros in Office so macro viruses have lost their popularity.

4. Third Wave: Mass E-Mailers

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.