Donald Wheeler & Luke Maraia

Professor Wu

SRA 221 Information Security

28 April 2025

Polymorphic Worms

Computer worms are a type of malicious software that exploit network connections between devices to propagate themselves. Unlike other computer viruses, worms do not require user interaction with the computer to spread further. Once a worm is able to infect one device, it takes advantage of vulnerabilities in common network protocols to copy itself to other devices. Worms can rapidly infect a large number of networked systems, spreading across both local networks and the greater Internet. Worms were created shortly after the conception of the Internet, originally as a way of experimenting with the capabilities of large networks. As the Internet has grown more sophisticated, so too have computer worms, as they are an effective method to exploit an ever-increasing number of networked devices across the world.

The first instance of a computer worm was named ‘Creeper’, originally developed by Bob Thomas of Bold, Beranek & Newman (BBN). BBN helped innovate the early packet-switched networks like ARPAnet that eventually led to the Internet. Bob Thomas developed the Creeper program as an experiment to see if a piece of computer software could propagate itself throughout a network without the need for human interaction. The Creeper software was developed for the TENEX operating system and moved between network devices running this OS. The first iteration of ‘Creeper’ is not technically considered a worm, as it only moved throughout networked devices rather than copying itself. A later iteration of Creeper implemented by programmer Ray Tomlinson did copy itself between devices, and this version is widely considered the first true computer worm. The program was not malicious, as it simply printed the message “I’M THE CREEPER : CATCH ME IF YOU CAN” to an affected device’s terminal. Ray Tomlinson later created the program ‘Reaper’, which functioned to catch Creeper by spreading across the ARPAnet and deleting instances of the virus. Reaper is considered by many the first instance of anti-virus software, but itself is also a virus by definition.

The first major attack on the Internet, according to the FBI, is the Morris worm. This worm was designed to target computers running the Berkely Unix OS, taking advantage of multiple software vulnerabilities in order to spread. One such vulnerability was in the ‘sendmail’ program, allowing the worm to propagate to other computers over the SMTP protocol, the Internet’s standard for email communication. The other major vulnerability was in the ‘fingerd’ program, a utility that can be used to find out information on network users. The program was vulnerable to buffer overflow exploitation. This allowed the worm to pass a parameter to the ‘fingerd’ program that overran the input buffer so that the return address could be overwritten to point to malicious code. According to the FBI, 6,000 computers were infected within 24 hours, amounting to about 10% of the Internet at the time. The worm was programmed to check if it had already infected a host but would copy itself anyways a percentage of the time. This led to computers being reinfected at a high rate, with the resources eaten by the virus inadvertently causing denial of service. The virus is estimated to have caused anywhere from $100,000 to $10,000,000 in total economic damage. It was discovered that the creator of the program was Cornell graduate student Robert Morris, a talented computer scientist. Morris became the first to be charged under the 1986 Computer Fraud and Abuse Act. The incident was a wake-up call to the U.S., revealing how vulnerable computers were and leading to major efforts to improve cybersecurity.

Early commercial antivirus software developed as a response to the emerging threat of potent computer viruses. These early products relied on signature-based detection to spot and remove malware. Signatures can identify a file or the contents inside of it. Antivirus software would compare the signatures of files on a system against known malware to detect infections. In 1990, Mark Washburn wrote the ‘1260’ virus, also known as ‘V2PX’, in an effort to demonstrate the limitations of contemporary antivirus software. It was derived from the source code of the 1987 ‘Vienna’ virus that targeted .com files on systems running DOS. The 1260 virus added a cipher and used a randomized decryption algorithm to vary its signature. This virus was the first known instance of a polymorphic virus.

A polymorphic virus is an instance of malware that uses cryptography to mutate its code and other identifiable features upon propagation. An attacker will often encrypt the payload of a virus so that it can bypass detection. Once the virus has passed security checks and been installed, the file is decrypted and executed. Polymorphic viruses rely on mutation engines to alter the contents of the payload upon decryption. The mutation engine randomizes identifiable characteristics of the file including the file name, type, size, location, and the encryption keys used. The code itself is also changed- variable names and order of operations are altered, and inconsequential ‘dead code’ is inserted. These changes further differentiate the file without changing the function or operation of the malware. The signature of the virus will change upon every new infection, meaning it will evade signature-based antivirus solutions.

An example of a polymorphic worm attack is the “Storm Worm” attack, which happened in 2007. This polymorphic worm attack targeted Windows operating systems, at its peak it infected an estimated one million computers. This virus combined the properties of a trojan, a bot, and a polymorphic worm. The way it spread was sent through a spam email that was titled

“230 dead as storm batters Europe.” This heading lured people to click it because at the same time Cyclone Kyrill was damaging parts of Europe. Once a user had been infected, the worm would inject a malicious service called wincom32.sys, which would open UDP ports for peer-to-peer communication, that would add the machine to the botnet. Injected machines would then be activated to download executable files over the Storm network to install backdoors, remote access, spreading the worm even further, or denial-of-service attacks. The Storm package used a packer that would then change the signature every ten to thirty minutes. No one is clear on who did it to this day, but Russian hacker group Zhelatin Gang were credited with this massive attack. Zhelatin gang is a notorious cybercrime organization that is based out of St. Petersburg, Russia.

There are many ways of defending against polymorphic worms. Firstly, updating your software and operating systems. Keeping these two up to date will eliminate holes that worms use to spread. Worms take advantage of bugs and weaknesses in operating systems; by just simply patching these holes up, it decreases worms' chances of spreading. Another way to defend against polymorphic worms is network segmentation. Rather than having all of the operating systems on one network, divide the network into zones. If a worm infects one zone, it will be much more difficult to reach the other zones. Similarly, automated containment will also help in stopping worms. If a system is portraying that it may be infected, an automated system that isolates the infected host will stop the worm from spreading. In addition, endpoint detection and response are an important layer of defending against polymorphic worms. If a worm is detected in a system trying to copy itself, the endpoint detection and response will quickly alert the security team to stop the worm.

Heuristic analysis is important to incorporate into antimalware solutions for the purpose of detecting polymorphic worms. Heuristic-based detection techniques spot malware by looking for suspicious behavior across systems, files and networks. This type of detection is adaptable and finds previously unknown or modified viruses that signature-based detection will miss. Heuristic analysis evolved to deal with the explosion in malware development as computer networking became ever more important at the turn of the century. There are several methods by which heuristic analysis functions. Static heuristic analysis decompiles software and inspects the source code, comparing it with known malware in a heuristic database. Software is flagged if a specified percentage of its code matches any known malware. Dynamic heuristic analysis uses a virtual machine to test software in an isolated environment. The software is executed to simulate its behavior and analyzed by antivirus software to look for suspicious activity. Heuristic-based detection methods are one of the few ways to reliably deal with polymorphic worms but must be fine-tuned or they will generate excessive false positives.

In summary, computer worms have evolved from networking experiments to potent cybersecurity threats, utilizing properties like polymorphism to evade detection and rapidly propagate. It is often human error that leads to an initial infection, but just one error can snowball into devastating effects. Computer worms eat up system resources and can have huge economic consequences as demonstrated by the Morris worm. The Storm Worm showed how sophisticated threat actors can use polymorphic worms to evade antivirus and launch massive cyberattacks. Antimalware solutions that use traditional signature-based detection need to adapt heuristic analysis methods to detect polymorphic viruses. Heuristic-based detection, patch management, network segmentation, and endpoint-based detection are all crucial elements of a cybersecurity program resilient to these sophisticated viruses. It is important to understand how polymorphic worms work and how to combat them as computer networking and malware continue to evolve in tandem, and as the world becomes increasingly reliant on digital infrastructure.

Works Cited

https://www.digitalguardian.com/blog/what-polymorphic-malware-definition-and-best-practices-defending-against-polymorphic-malware

https://www.exabeam.com/blog/infosec-trends/creeper-the-worlds-first-computer-virus/

https://www.digitalguardian.com/blog/what-polymorphic-malware-definition-and-best-practices-defending-against-polymorphic-malware

https://blog.barracuda.com/2023/06/06/malware-101-worms

https://www.techtarget.com/searchsecurity/definition/worm#:~:text=A%20computer%20worm%20is%20a,to%20spread%20to%20uninfected%20computers

https://www.fbi.gov/news/stories/morris-worm-30-years-since-first-major-attack-on-internet-110218

https://www.secureworks.com/research/storm-worm

https://www.sentinelone.com/blog/what-is-a-malware-file-signature-and-how-does-it-work/

https://www.crowdstrike.com/en-us/cybersecurity-101/malware/polymorphic-virus/

https://www.informit.com/articles/article.aspx?p=366890&seqNum=5#:~:text=The%20first%20known%20polymorphic%20virus,junk%20instructions%20into%20its%20decryptor.

https://usa.kaspersky.com/resource-center/definitions/heuristic-analysis?srsltid=AfmBOooaJ03Oqtmx3Mrwn_XpJvvH3AXC0TLxHZxjh1NF-DtpNkam9yk8