The Morris Worm Incident of 1988

Teaundra Thompson

Charleston Southern University

CSCI 405: Principles of Cybersecurity

Professor Hill

September 16th, 2025

Abstract

This paper details the creation, and actions of the infamous 1988 Morris Worm created by Robert Tappan Morris Jr. The actions taken by the worm and the mitigation steps to stop it will be discussed. The worm itself was not advanced, but the sheer number of times it replicated itself was. The internet was temporarily out of commission due to the number of computers that were infected. The machines infected with the worm were ten percent of the computers connected to the internet in 1988. This unprecedented attack left the early internet in shambles and caused international chaos amongst universities and governments. This attack created precedence for cyber-attacks in the coming future and introduced new legislation regarding the use of computers and technology. Robert Morris faced legal repercussions for his actions, but his actions also fostered awareness for the need for cyber security.

Keywords: Morris Worm, internet, security, computer, machine

The Morris Worm Incident of 1988

The Morris Worm's unprecedented release in 1988 changed the entire landscape of the internet and helped morph it into the "World Wide Web" that we have today. Robert Tappan Morris Jr. was a student at the prestigious Cornell University and the culprit that naively released the malicious worm to the internet. Robert's intention was to create something that could measure the current size of the internet, but the whole project unraveled into a catastrophe due to the nature of the code (Harán, 2018). The havoc that the worm wreaked on the early internet illuminated numerous security holes and brought awareness to cybersecurity. Robert Jr. was a trailblazer of sorts due to him being the first individual to be convicted under the Computer Fraud and Abuse Act.

The Worms Release

The first computer worm in history was released into the internet on November 2nd of 1988 (Orman, 2003). It was estimated that one in ten of the 60,000 computers that were connected to the internet were infected with the worm (*The Morris Worm*, 2018). A cyber-attack of this scale was unprecedented, and it affected computers connected to the internet on an international scale. Every government, college, and university were sent scrambling to mitigate the damage of the worm. Not only was the worm the first malware attack, it also was the first DDoS attack (Vaughan-Nichols, 2018). The worm brought the internet to a slow crawl and crashed countless computers by continuously reinfecting machines and searching for new victims. The code for the program was configured to execute on VAX computers running BSD UNIX and also Sun 3 computers (Spafford, 1989). Notably, four years prior to the release of the worm, Morris co-authored an article, "*The* UNIX *System*: *UNIX* Operating System Security" which highlighted the security hazards of the UNIX operating system and suggested ways to protect against them (Grampp & Morris, 1984). One can speculate due to the nature of the aforementioned article; he created and released the worm to highlight exactly how vulnerable the UNIX operating system truly was. Robert Morris released the worm from a computer at the Massachusetts

Institute of Technology by remotely hacking into the computer from his terminal at Cornell University (*The Morris Worm*, 2018). The Morris Worm was supposed to be a discrete experiment, but it quickly spiraled out of control by rapidly infecting computers connected to the internet by exploiting a backdoor in the sendmail email service and a bug in the fingerd program. To prevent further spread of the worm, universities and government agencies disconnected from the internet altogether (Disrupt, 2021). Upon realizing just how much the worm had spun out of control, Robert Morris had a friend to relay a message across the internet with an apology and instructions for removing the program, but only a few amount of people received the message because the internet was bogged down with the constant replication of the worm (*The Morris Worm*, 2018). Another friend of Robert Morris reached out to *The New York Times* and told them he knew the creator of the destructive program, and it was intended for a harmless prank, but a small programming error resulted in its rampant spread. The friend mistakenly referred to the creator by his initials, RTM, and from there *The New York Times* confirmed and reported that Robert Tappan Morris Jr. was indeed the culprit behind the first internet worm.

The Inner-Workings of the Worm

A computer worm is commonly labeled as a virus, and it is important to note the distinct characteristics when discussing the Morris Worm. A virus is a piece of code that needs to add itself to an existing program to run; on the other hand, according to Spafford (1989, p. 3), a worm is:

> "… a program that can run by itself and can propagate a fully working version of itself to other machines. It is derived from the word tapeworm, a parasitic organism that lives inside a host and saps its resources to maintain itself."

This definition perfectly describes the behavior of the Morris Worm. It was a program that was self-sustaining and it rapidly replicated itself and distributed its copies to other machines. It also sapped the memory of infected machines causing them to crash. The Morris Worm had two parts: the main program, and a vector program. In the following description of the worm's attack sequence, it is described with the notion that the worm successfully infected a host machine and is now transmitting itself to another machine. The worm begins its attack by burrowing itself into a motherboard and creating its own socket. The vectors program is subsequently installed and executed by either utilizing a TCP connection to a shell or a SMTP connection to the previous worm. Once installation and execution are confirmed the vector program connects to the server worm and copies three files: a SUN 3 binary version of the worm, a VAX version, and the source code. If one version of the worm binary file failed to work, the program would try the next one. If both files failed the worm would send "rm" commands for the files to be deleted to erase evidence of the infection attempt. Once the server worm confirmed that the host was infected it would then close the connection. The worm then chooses one of three methods to continue its burrowing. One of the methods utilized is the fingerd protocol, which drew parallels to a social network due to the fact it housed usernames and contact information from individuals who accessed the internet (Disrupt, 2021). Fingerd had a flaw; it used an old library call "gets" that did not verify the size of the input buffer (Computerphile, 2020). The worm would then intentionally overflow the buffer which allowed it to start overwriting code on the stack, and the worm would then run a command interpreter and successfully reveal other users on the internet. A second method utilized was remote shell. The worm would look for accounts set up for remote shell and would utilize it to jump to other machines. The final method used was sendmail. Sendmail had a backdoor with the "debug" mode. If debug mode were enabled the worm could send commands to execute itself. When the worm got access to a machine it would brute force user accounts by utilizing a list of possible passwords. If the worm successfully guessed the user credentials it would then check if there were any remote machines linked to the account and infect them also (Orman, 2003). There was a critical flaw in the worm's source code that made it more aggressive than intended. The worm would repetitiously reinfect machines, and one out of every seven worms would never self-destruct even if the host were already infected (Spafford, 1989). This is what led to computers overloading and crashing, making Robert Morris' "discrete experiment" loud and pernicious. Notably, worms that were marked to self-destruct did not do so until they made one complete pass through the password file. If Robert Morris had not overlooked this error, his program would have most likely gone undetected and his experiment would have succeeded. Even if his experiment had succeeded, there would be major ethical issues due to the worm gaining unauthorized access to computers without prior knowledge and approval.

The Aftermath of the Worm

Although the Morris Worm caused chaos throughout the internet, the modern internet and modern machines benefitted greatly from it. The Morris Worm exploited several security vulnerabilities within the internet and networking protocols. If Robert Morris had not released the destructive worm, someone else would have done it and their intentions would possibly be truly nefarious. For initial mitigation many organizations simply unplugged from the internet to prevent further infection (Computerphile, 2020). Late night of November 2, 1988 copies of the worm program was captured by staff at the University of California, University of Berkeley, and Massachusetts Institute of Technology; they were able to disassemble the code to determine what it was doing and created solutions (Spafford, 1989). In the early morning of the next day a research group at Berkeley released patches for the sendmail program that prevented the program from accepting the "debug" command and compiling with the worm (Disrupt, 2021; Spafford, 1989). Purdue University released the next patch; this patch fixed fingerd so that it would no longer accept the "gets" call and instead only accept "fgets" which prevented buffer overflow. A quick "hack" that was found to stop the worm was to create a directory named "user_thump_sh" (Computerphile, 2020). Creating this directory killed the worm's process because the worm would try to write to a temporary file "user_thump_sh," but it did not check if there was a directory with the same name. If there was a directory with the same name the worm would no longer be able to continue and would self-delete. After Robert Morris was identified as the author of the worm he was then placed on leave from Cornell University due to him violating their policies regarding “computer abuse” (Eisenberg et al., 1989). Robert Morris also violated the federal Computer Fraud and Abuse Act (CFAA) that was passed by Congress in 1986 (*The Morris Worm*, 2018). This law prohibits unauthorized access to a federal computer and preventing authorized use of that machine (Shemakov, 2019). Morris was indicted in 1989 and was found guilty in 1990, making him the first person to ever be convicted under the 1986 act. Fortunately for Morris, he received a fine of $10,050, three-year probation, and was required to complete four hundred hours of community service. This conviction opened the door for more legislation to be made concerning the use of technology and consequences for infractions. Representative Wally Herger amended the CFAA with the Computer Virus Eradication Act (CVEA). The amendment seeked to include private firms’ networks into the CFAA’s scope, not just federal. Several states also enacted laws making the release of computer viruses a crime. The Defense Advanced Research Projects Agency (DARPA) created the Computer Emergency Response Team (CERT) which serves as a clearing house for security-vulnerability information (Orman, 2003). CERT is responsible for reporting incidents, managing security research, and releasing alerts regarding security issues (Shemakov, 2019). Another significant improvement that arose was firewalls, which are vital for isolating private networks from the internet to block malicious traffic to prevent harm to internal systems. Many private firms marketed this concept and incorporated firewalls into their commercial anti-virus and computer security systems.

The Conclusion

The Morris Worm helped to mold the modern internet by bringing awareness to easily exploitable internet protocols. It brought international awareness to the need for cybersecurity and good security practices. A plethora of legal and ethical guidelines were created in the wake of the worm and it created precedent for other cyber crimes to come in the future. The worm also contributed to a culture of “hacking” and “exploitation.” This hacking culture also contributes to the ever-growing field of cybersecurity. There is a vital need for cyber security, the same way there is a dire need for law and order within our societies. Without law and order within the cyberspace it would be chaotic just like when the Morris Worm was released. Even though there are dark and morbid corners of the internet, such as the dark web, those areas are confined to a miniscule segment of the internet and activities there are monitored by government agencies. The early internet had a plethora of fallacies, but the Morris Worm appearance quickened the mitigation and resolution of some of those fallacies. Robert Morris' method of conducting his experiment and his lack of accountability when first realizing it spun out of control was unethical and unprofessional. Despite his lack of ethical behavior, his work was tremendously beneficial because universities and government agencies sprung into action and collaborated to create policies and secure networking policies and established precedence for similar cyber-attacks.

References

Computerphile. (2020, October 30). *The First Internet Worm (Morris Worm)—Computerphile* [Video]. https://www.youtube.com/watch?v=2QwMv0_Rkec

Disrupt. (2021, December 1). *MORRIS: Earth's First Computer Worm* [Video]. https://www.youtube.com/watch?v=7BRTixIDzzE

Eisenberg, T., Gries, D., Hartmanis, J., Holcomb, D., Lynn, M. S., & Santoro, T. (1989). The Cornell commission: On Morris and the worm. *Commun. ACM*, *32*(6), 706–709. https://doi.org/10.1145/63526.63530

Harán, J. (2018). *Malware of the 1980s: Looking back at the Brain Virus and the Morris Worm*. Retrieved September 11, 2025, from https://www.welivesecurity.com/2018/11/05/malware-1980s-brain-virus-morris-worm/

Grampp, F. T., & Morris, R. H. (1984). *The* UNIX *System: UNIX* Operating System Security. *AT&T Bell Laboratories Technical Journal*, *63*(8), 1649–1672. https://doi.org/10.1002/j.1538-7305.1984.tb00058.x

Orman, H. (2003). The Morris worm: A fifteen-year perspective. *IEEE Security & Privacy*, *1*(5), 35–43. https://doi.org/10.1109/MSECP.2003.1236233

Shemakov, R. (2019). *The Morris Worm: Cyber Security, Viral Contagions, and National Sovereignty*. https://works.swarthmore.edu/theses/544

Spafford, E. H. (1989). The internet worm program: An analysis. *ACM SIGCOMM Computer Communication Review*, *19*(1), 17–57. https://doi.org/10.1145/66093.66095

The Morris Worm. (2018). [Story]. Federal Bureau of Investigation. Retrieved September 11, 2025, from https://www.fbi.gov/news/stories/morris-worm-30-years-since-first-major-attack-on-internet-110218

Vaughan-Nichols, S. (2018). *The day computer security turned real: The Morris Worm turns 30 | ZDNET*. Retrieved September 11, 2025, from https://www.zdnet.com/article/the-day-computer-security-turned-real-the-morris-worm-turns-30/