EternalBlue: How an NSA Leak Crippled One of the World's Most Important Operating Systems

Joshua Miller
Old Dominion University
CYSE 280
Professor Malik Gladden
Fall 2023

Microsoft Windows is one of the most popular operating systems in use today. It fills a niche in the home, server, and workplace that no other operating system has been able to fill. What happens when an enterprise operating system that dominates the market is compromised by elite hackers? What happens when these "hackers" are the most sophisticated and advanced security professionals in the world? This is the case for the exploit MS17-010, otherwise known as EternalBlue, an exploit that crippled systems that used Microsoft Windows.

On the morning of May 12th, 2017, the first incident in a long succession of cyberattacks had been inflicted on a Spanish telecommunications company. From here, the attack would spread to other organizations and networks, one of which being the UK's National Institute of Health (NIH). The NIH would be the most heavily affected target of this attack, and by the afternoon of the 12th, many of the windows machines that were connected to each other would be affected by ransomware.

Ransomware is a type of malware that, after infecting computers, encrypts their files and demands a "ransom" be paid out, with instructions on screen to pay out the ransom. In the case of the May 12th cyberattack, this wasn't a bluff, as files of the computer, barring ones required for core functionality of the operating system, were encrypted, unreadable, and only recoverable with the decryption key. To the horror of the NIH, the ransomware had targeted computers that were critical to the hospital's operation. While systems were offline, doctors, nurses, and other medical personnel had to find alternate ways to make and view records, test patients, and process patients.

In the meantime during this outage, cybersecurity experts from around the world were taking action to find out where this attack came from, how it spread, how to recover the files, and most importantly, how to stop it. Both independent researchers and security teams had started to

make efforts to procure their own “lab sample” of the WannaCry ransomware to research. If the code could be examined, maybe researchers could find ways to stop the malware from spreading. Many large companies, like FireEye and Microsoft, went at this with all of their might in order to quickly find a solution, but it seemed to researchers that this malware was only affecting older versions of Windows that had not been patched.

While the researchers worked overtime to find a solution to this problem, the WannaCry malware was spreading rapidly. Over 150 countries reported cyberattacks that resembled the attacks that crippled the NIH, with hospitals, government offices, and even universities being affected. It was clear that the longer this threat was out in the wild, the more catastrophic it would become.

When a 22 year old freelance cybersecurity researcher named Marcus Hutchins had gotten his hands on a sample of the Wannacry malware, he began to analyze it. It was here that he realized some of the inner workings of the malware and the sequence of events that would unfold in the code as the payload was being executed. He realized that after infection of the computer, but before the release of the payload and encryption of files, the malware would ping a certain domain before continuing with the encryption process.

The domain “iuqerfsodp9ifjaposdfjhgosurijfaewrwergwea.com”, an apparent random string of characters with a “.com” attached to the end of it, was being pinged before further operation. Hutchins wanted to figure out the function of this domain, as it wasn’t apparent in WannaCry’s code what it was for. After visiting a domain registry to attempt to find the owner of this domain, it became apparent that the domain wasn’t registered at all. Hutchins made an unprecedented (and even risky) move and registered the domain.

From this point on, the attacks around the world appeared to stop. WannaCry, after attempting to reach this newly registered domain, would then abort its operation. Computers would still be “infected” with the malware, but the encryption would no longer take place. The actions of this lone cybersecurity researcher had stopped a cyberattack that had the potential to cripple the entire world.

There is no clear consensus as to the purpose this unregistered domain had. Some believe the domain functioned as a potential remote “killswitch” that could be flipped in case the attack had gotten out of hand. More recent theories conclude that the pinging of the unregistered domain functioned as a sort of sandbox detection. A sandbox is an isolated instance of a machine, application, or program that is contained on a machine. If an instance of malware were to ping a domain on a sandbox that is not connected to the internet, the sandbox would be able to detect the function and false queries from the domain could be spoofed back to the machine. Since the domain was not registered (and due to the random nature of the domain name, would never be accidentally registered), the malware would know that it is in a sandbox and would end its processes.

Now that the threat was neutralized, cybersecurity researchers could attempt to reverse engineer this malware to see what the inner workings entailed. While computers that had been encrypted were not decrypted by this killswitch, the computers that had not been encrypted retained full functionality. Some analysis of this malware showed that the exploits EternalBlue and DoublePulsar had been used in the WannaCry malware. These exploits were known to Windows and had been patched out of their more recent builds of Windows. Older builds, such as those used in hospitals and businesses needing stable functionality from their machines, had

not patched their systems. This led Microsoft to release emergency patches for older systems, something which is rarely done.

What made Wannacry such a formidable piece of malware? There have been ransomware variants that have attacked businesses and even hospitals before, but not to the scale of the WannaCry attack. WannaCry wasn't just a ransomware, but a ransomware that propagated easily and automatically with remote code execution capabilities. The exploit that made WannaCry such a damaging piece of malware was named EternalBlue, which had a propagation potential never before seen and a surprising history.

EternalBlue is a Windows exploit that affects the early versions of the Server Message Block (SMB) protocol. SMB, which is a file sharing protocol that allows machines to share files with each other within a network. This is integral for many networks and servers in businesses and other network infrastructure. Both versions 1 and 2 of the SMB protocol were susceptible to the EternalBlue attack, with TCP 445 (Microsoft Domain Service) and TCP 139 (NetBIOS Session Service) transmission protocols specifically being affected.

Server Message Block data is contained in packets known as NT Trans(mission) packets that are established after the SMB session is created. It is often required for more than one packet to be used to transmit data, depending on data size. Instead of sending an NT Trans 1 packet over and over again, an NT Trans 1 and 2 packet will be sent together to encapsulate all of the data. The amount of data that can be contained and sent in an NT Trans packet is controlled by a value known as the SMB Max Buffer Size within the SMB protocol.

The exploit EternalBlue abuses a memory overflow, a common anomaly which is the basis for many code bugs and security exploits. NT Trans packets need to be specifically configured or "formed" for the data transfer process, and EternalBlue exploits this configuration

and creates “malformed” NT2 packet headers. As discussed earlier, NT Trans 2 packets take over when NT Trans (1) packets are “full”. EternalBlue pads the NT Trans 1 packets with zeros to initiate the introduction of the NT Trans 2 packet and uses a malformed header with a pointer to a section of the variable memory. Pointers to variable memory are not secure, and this overflow of memory allows for the introduction of the shellcode and payload, which in this case is the DoublePulsar exploit.

EternalBlue isn’t the only part of the WannaCry malware, but it is the exploit that does the heavy lifting and enters the defenses of the Windows machine, at which point DoublePulsar takes over and exploits the overflowed memory. Memory overflows in the C language, which the affected parts of Windows were written in, are common, but are oftentimes hard to detect. To discover a memory exploit in this way takes a skilled computer scientist or cybersecurity engineer, which in the caliber previously described are rare.

The NSA, or National Security Agency of the United States, are the defacto cybersecurity experts of the world. They employ the world’s most skilled computer scientists, mathematicians, and cybersecurity experts in the world in order to assist the United States’ goal of global security. After the release of NSA created exploits such as those of the Eternal Family (EternalChampion, EternalRomance, and EternalBlue), much discussion around the creation and stockpiling of said weapons took place between software vendors like Microsoft.

When an exploit is discovered and stockpiled for use later by a government agency such as the NSA, it can be considered a cyber weapon. While the concept of a government stockpiling cyberweapons exists in a moral gray area, it can (and did, in this case) have repercussions on society. In August of 2016, these NSA “tools” as they were called, appeared on the black market. EternalBlue was among these tools. On April 14th of the following year, these tools were

released on Github by a group that became known as the Shadowbrokers. From here, the exploit was officially in the wild, all stemming from a leak from the NSA.

EternalBlue was one of the most dangerous cyberweapons ever seen by the world. Its effectiveness came not from the destruction it would initially cause the system, but by the systems it would affect, the exploits and payload it would come bundled with, and the skill involved in researching and creating such an exploit. EternalBlue exploited the memory of the SMBv1 and SMBv2 protocols of Windows, one of the most widely used operating systems on earth, and caused billions of dollars in damages to critical systems such as universities, government offices, and hospitals. While discussion has been had around the ethics of the NSA stockpiling cyber exploits to be used as weapons, one thing not in dispute is the harm that could have been avoided had the NSA disclosed these exploits quickly once leaked. Microsoft, although fast with their initial patching of the systems, could have and should have rolled out a patch, even for older systems that are not supported. Doing this, although in conflict with good "cyber hygiene" of always keeping systems patched with their latest software, would have helped legacy systems still running from the immediate attacks. While it is important to prepare for the future, cybersecurity professionals must also learn from the past, and with EternalBlue, we have a lot of lessons to learn.

Works Cited:

Aiden, J S, et al. “Comprehensive Survey on Petya Ransomware Attack.” *2017 International Conference on Next Generation Computing and Information Systems (ICNGCIS)*, 11 Dec. 2017, pp. 122–125, https://doi.org/10.1109/ICNGCIS.2017.30.

Burdova, Carly. “What Is EternalBlue and Why Is the MS17-010 Exploit Still Relevant?” *What Is EternalBlue and Why Is the MS17-010 Exploit Still Relevant?*, Avast, 23 Feb. 2023, www.avast.com/c-eternalblue.

“Cyberattack Hit More than 100,000 Groups in at Least 150 Countries, Europol Says.” *CBS News*, CBS Interactive, 14 May 2017, www.cbsnews.com/news/cyberattack-hit-more-than-100000-groups-in-at-least-150-countries-europol-says/.

Greenberg, Andy. “The Confessions of the Hacker Who Saved the Internet.” *Wired*, Conde Nast, 12 May 2020, www.wired.com/story/confessions-marcus-hutchins-hacker-who-saved-the-internet/.

Gupta, Manoj. “EternalBlue Vulnerability .” *Researchgate.Net*, June 2023, www.researchgate.net/publication/371470741_Eternal_Blue_Vulnerability.

Islam, Ali. “SMB Exploited: WannaCry Use of ‘Eternalblue.’” *Mandiant*, 26 May 2017, www.mandiant.com/resources/blog/smb-exploited-wannacry-use-of-eternalblue.

Liu, Zian, et al. “Working Mechanism of eternalblue and its application in ransomworm.” *Cyberspace Safety and Security*, 2022, pp. 178–191, https://doi.org/10.1007/978-3-031-18067-5_13.

Palmer, Danny. “Leaked NSA Hacking Exploit Used in WannaCry Ransomware Is Now Powering Trojan Malware.” *ZDNET*, 5 June 2017, www.zdnet.com/article/leaked-nsa-hacking-exploit-used-in-wannacry-ransomware-is-now-powering-trojan-malware/.

Rhysider, Jack. “Darknet Diaries.” Season WannaCry, episode 73.

Robert Lemos, Contributing Writer. “Eternalblue Longevity Underscores Patching Problem.” *EternalBlue Longevity Underscores Patching Problem*, 17 Oct. 2023, www.darkreading.com/vulnerabilities-threats/eternalblue-longevity-underscores-patching-problem.

Shivanandhan, Manish. “Eternalblue Explained – an in-Depth Analysis of the Notorious Windows Flaw.” *freeCodeCamp.Org*, freeCodeCamp.org, 11 Sept. 2023, www.freecodecamp.org/news/eternalblue-explained-an-analysis-of-the-windows-flaw/.

Smith, Brad. “The Need for Urgent Collective Action to Keep People Safe Online: Lessons from Last Week’s Cyberattack.” *Microsoft On the Issues*, 14 May 2017, blogs.microsoft.com/on-the-issues/2017/05/14/need-urgent-collective-action-keep-people-safe-online-lessons-last-weeks-cyberattack/.

“SteelCon 2018 EternalBlue: Exploit Analysis And Beyond by Emma McCall.” SteelCon.

University, Stephen B. Wicker Cornell, et al. “The Ethics of Zero-Day Exploits---: The NSA Meets the Trolley Car: Communications of the ACM: Vol 64, No 1.” *Communications of the ACM*, 1 Jan. 2021, dl.acm.org/doi/10.1145/3393670.

“What Was the WannaCry Ransomware Attack? .” *Cloudflare Learning* , www.cloudflare.com/learning/security/ransomware/wannacry-ransomware/. Accessed 7 Dec. 2023.