NotPetya Ransomware: Technical Analysis and Investigation into its Influence on the Cyberthreat Ecosystem

Sean Carroll

ECE 8486: Ethical Hacking

Villanova University

Abstract

Ransomware attacks have seen a rapid growth in popularity since their appearance in 2012. Ransomware-as-a-Service (RAAS) and the use of ransomware in Cyberwarfare has help foster an environment in which ransomware attacks are highly advanced, targeted, and damaging. The technical analysis of previously detected ransomware threats is crucial to the prevention and mitigation of similar attacks in the future. This paper discusses the NotPetya variant of the Petya ransomware malware, including the technical methods of attack, the possible profiles and motives of the actors orchestrating the attack, and the implications it has on the ecosystem surround malware attacks. Remediation and mitigation strategies for ransomware-specific attacks are also proposed.

NotPetya Ransomware: Technical Analysis and Investigation into its Influence on the Cyberthreat Ecosystem

The NotPetya ransomware appeared on in June 2017 through a vulnerability in a Ukrainian tax accounting software called MEDoc. Malicious code was injected through a remote software update on network computers. Upon reboot, the Master File Table (MFT) is encrypted, rendering the Master Boot Record (MBR) of the operating system unusable (Fayi, 2018). NotPetya spreads to operating systems over the network through use of vulnerabilities EternalBlue, EternalRomance, PsExec, Windows management instrumentation, and Admin$ shares. Additionally, Notpetya spread phishing emails and the implementation of mimikatz, an open-source tool used to collect windows user credentials of other hosts on the network (Noerenberg, 2017). NotPetya originally starts as a Windows DLL with the file name "perfc.dat," but after defining itself as a ransomware virus, begins to exploit attack vectors that are commonly seen in network worms (. Due to the complexity of the attack, the flow of commands is shown in figure (1), which represents a modified version of Cynet's original technical analysis.

To consider the NotPetya attack a variant would be to assume that it derived its code from the base Petya attack. Sood, K., and Hurley, S's (2017) technical analysis of NotPetya has shown that although similar in design, the "NotPetya" variant was not a modification of the original Petya, but a re-implementation of similar code provided by the GoldenEye malware source code. Online leaks from political activist or political espionage groups allowed for the core code to create similar source codes from the original Petya code. Many technical analyses show that the binary of Petya was used in the attack, however, a hex editor was used to edit certain parts of the

program to change the user interface and to prevent certain services of the original binary source code from running.

As with roughly 80 percent of malware threats on the internet, we see a reuse or repackaging of malware from malicious software that has already been produced (Bat-Erdene, et al., 2017). However, current trend analysis has seen a much greater increase in targeted attacks, as in ones with a specific purpose and a specific group in mind for mode of action. The NotPetya could appear to be an amalgamation of various reused codes, with some of the most logical routes not necessarily taken, but reverse engineered to efficiently conduct the malicious intent desired. Within the malware, we see that there is a decision tree-type code that exists, which checks for specific anti-virus and then makes decisions based on their presence on the infected host operating system. The oddity that arises with this case statement-type check, is that it provides a potential kill switch for itself given certain conditions on the system (Noerenberg, 2017). The intent of the programmers must not have been to retrieve the ransom paid, but to wipe data from host and network systems.

Discussion of the Actors and their Influence

When considering the advisories of the attack, it is very unclear and subjective as to who truly conducted the attack. We must consider that the authors of malware attacks are very well-versed in the art of deception. The code appears to have some parts that are reused from other malware, with other parts looking very poorly written, some features that could have been enabled, but failed on execution. Furthermore, other parts of code that were implemented demonstrated a advanced understanding of computer hardware, the Microsoft operating systems, and the entire full-stack of libraries provided by Microsoft. With the varying levels of knowledge, the profile of the creator or creators fits into one of an older-aged software

developer, with great knowledge of the Microsoft platform, yet on a time restraint. We also have to consider that the way in which the code was presented may be an identity that the creators intended to portray by design, in order to mislead research analysists. The current profile fits the possibility that Russia could have created the attack as a form of cyberwarfare, however, attribution is not certain.

The tie to the Petya malware may have been a form of obfuscation of the actual intent of the creators of the NotPetya malware as well. The tie to other variants and the various other Petya variants may have been intending to hide the novelty and intent of the NotPetya attack by tying similar attacks with different motives and different actors to cause misinformation.

We also must consider the way the malware was spread, and the political players in the field that we see today. The ShadowBrokers leak, as well as the political atmosphere in eastern European countries, such as Ukraine at the time of the attack, put many different factors into the situation that create greater plausible perpetrators. Several countries, including the United States and the United Kingdom, accuse Russia of designing the targeted attack (Fayi, 2018). The area in which the vulnerability was leaked, however, is a very common testbed for malware in general, with various countries originating their attacks in Eastern Europe.

The NotPetya attack caused an estimated 10 Billion dollars and damage to companies since its implementation at the time of the writing of this paper (Greenberg, 2018). Let us not consider the intent or the adversarial group at hand but examine the possible intent and repercussions for future infrastructure operations of large systems with dated and cutting-edge infrastructures connected through various means. Because the NotPetya was not necessarily a ransomware, as it did not collect a substantial ransom, but only wiped compromised data, the future implications of such attacks is of greatest interest due to NotPetya's novel attack method.

Future actors could use this attack to potentially command and control losses at firms through attacking their dated infrastructure networks. The NotPetya could also be a test-drive for an application that attempts to force industry to adopt and upgrade more-secure and updated systems on their networks. A future attack could potentially attack dated SQL databases, or dated software implementations to force the upkeep of commercial infrastructure, through forceful means. It is often the case that in regard to regulatory mandates, companies wait until they are absolutely required to implement the new security standards. The earning capacity of investment over that wait time allows for more financial gain yet grows the attack surface in which could be potentially exploited.

Speculation of the NotPetya being a cover-up of a different attack on national infrastructure that did not want to be exposed also proposed another possibility and could explain why the malicious software acted as a wiper (Greenberg, 2017). It should be noted that the NotPetya specifically targeted the Master-Boot Record (MBR) and MFT (Microsoft File System) on disks. Although the current Windows 10 operating system can be partitioned to MBR, GPT using UEFI have now become the standard due to the numerous benefits. This is notable that the NotPetya attack chose to attack only dated partitions. Since this attack on 2017, we have seen several changes in default disk formats on major operating systems, including Mac OSX requiring only APFS on their most recent release, as well as the most recent Ubuntu moving away from swap partitions. It does appear that filesystem defaults are changing to combat the security issues with dated partitions schemes. The interoperability between SMB, NETBIOS, and MBR, is a thirty-year old design, where the software retrofitting of such protocols resulted in huge monetary loss (Gofman, 2017).

Mitigation and Preventative Strategies, Conclusion

As part of the mitigation strategy of such attacks demands the updating older hardware and software designs to reduce the attack surfaces on potentially vulnerable parties. Furthermore, the backup of data crucial to operation is an advised preventative measure. Several security solutions have touted their anti-virus software as mitigating these threats, providing detailed analysis on the malware in attempt to attacked customers. However, none of these measures is an all-encompassing solution to the prevent against attacks in the future.

In consideration of NotPetya, and, with the current state of the internet as a whole, an optimal approach for the problem of targeted Cyberthreats is not to focus on the intricate technical details of such threats and CVE's representing microcosms in the cybersecurity space. Certainly, those should be addressed and mitigated once they occur. However, to take a optimal approach to Cybersecurity and its attack vectors, we must view the current ecosystem as a macrocosm, assuming that internal factors and external factors within concerned environments are not necessarily under our control, despite measures taken to mitigate risk. An optimal approach to such threats is to seek a system engineering perspective in which we look at the entire space as whole, knowing that specific attacks may not be preventable.

There are several measures that could be put into place that would significantly reduce the probability of threat to specific public or private firms. The inclusion of both Host Intrusion Defense Systems and Network Intrusion Defense Systems is a common IT approach to monitor traffic on the network. There are also common preventable measures such as ongoing bug fixing on network computers and the education of staff on cybersecurity best practices. A mind map created by Sivaraj Ganesan is shown in figure (2) to list common best practices in solving ransomware attacks.

Oftentimes, with the onset of new technologies in industry, the demand for backwards integration is equally as important. The requirement of new technologies demands greater training, and greater burdened to those not technically inclined. The need for backwards compatibility poses as serious threat to the company's future growth, albeit the disruption and expense that would be required to bypass the backward compatibility is a major deterring factor. As older and older devices remain tied to the other network infrastructure, it creates increasing attack vectors that could potentially plague not only the antiquated systems, but the new, very secure systems through interconnectivity.

The current OSI model has many protocols on each layer, many having major with security flaws. However, the reality is that this will proceed into the future, despite patches, improvements, and new protocols, there will always be points of exploitation. The implementation of a macro designed solution is one that will ultimately be effective in thwarting unknown or unexpected vulnerabilities. The proposal of an entirely new model for cybersecure systems is not realistic from a large-scale perspective, thus, a disruptive and innovative approach is not a realistic consideration, given the incredible amounts of cost implied to reach such a solution. With specific regards to the rapidly growing ransomware malware family, it is the hope that this discussion spawns further research into efficient malware prevention and mitigation strategies.

References

Bat-Erdene, M., Kim, T., Park, H., & Lee, H. (2017). Packer Detection for Multi-Layer Executables Using Entropy Analysis. Entropy, 19, 125.

Carbon Black Threat Research Technical Analysis: Petya / NotPetya Ransomware. (2017, September 27). Retrieved from https://www.carbonblack.com/2017/06/28/carbon-black-threat-research-technical-analysis-petya-notpetya-ransomware/

Fayi, S. Y. (2018). What Petya/NotPetya Ransomware Is and What Its Remidiations Are. *Advances in Intelligent Systems and Computing Information Technology – New Generations,* 93-100. doi:10.1007/978-3-319-77028-4_15

Gofman, I. (2017, October 3). Advanced Threat Analytics security research network technical analysis: NotPetya. Retrieved from https://cloudblogs.microsoft.com/microsoftsecure/2017/10/03/advanced-threat-analytics-security-research-network-technical-analysis-notpetya/

Greenberg, A. (2018, August 24). The Untold Story of NotPetya, the Most Devastating Cyberattack in History. Retrieved from https://www.wired.com/story/notpetya-cyberattack-ukraine-russia-code-crashed-the-world/

Intezer Analysis of NotPetya Ransomware. (2017, July). Retrieved from http://www.intezer.com/wp-content/uploads/2017/07/Intezer-NotPetya.pdf

Noerenberg, E. (2017, June 30). NotPetya Technical Analysis | LogRhythm. Retrieved from https://logrhythm.com/blog/notpetya-technical-analysis/

NotPetya Ransomware analysis. (2017, July 03). Retrieved from https://safe-cyberdefense.com/notpetya-ransomware-analysis/

Petya ransomware outbreak: Here's what you need to know. (2017, October 24). Retrieved from https://www.symantec.com/blogs/threat-intelligence/petya-ransomware-wiper

Sood, K., & Hurley, S. (2017, June 29). NotPetya Ransomware Attack Technical Analysis. Retrieved from https://www.crowdstrike.com/blog/petrwrap-ransomware-technical-analysis-triple-threat-file-encryption-mft-encryption-credential-theft/

NotPetya Flow Chart

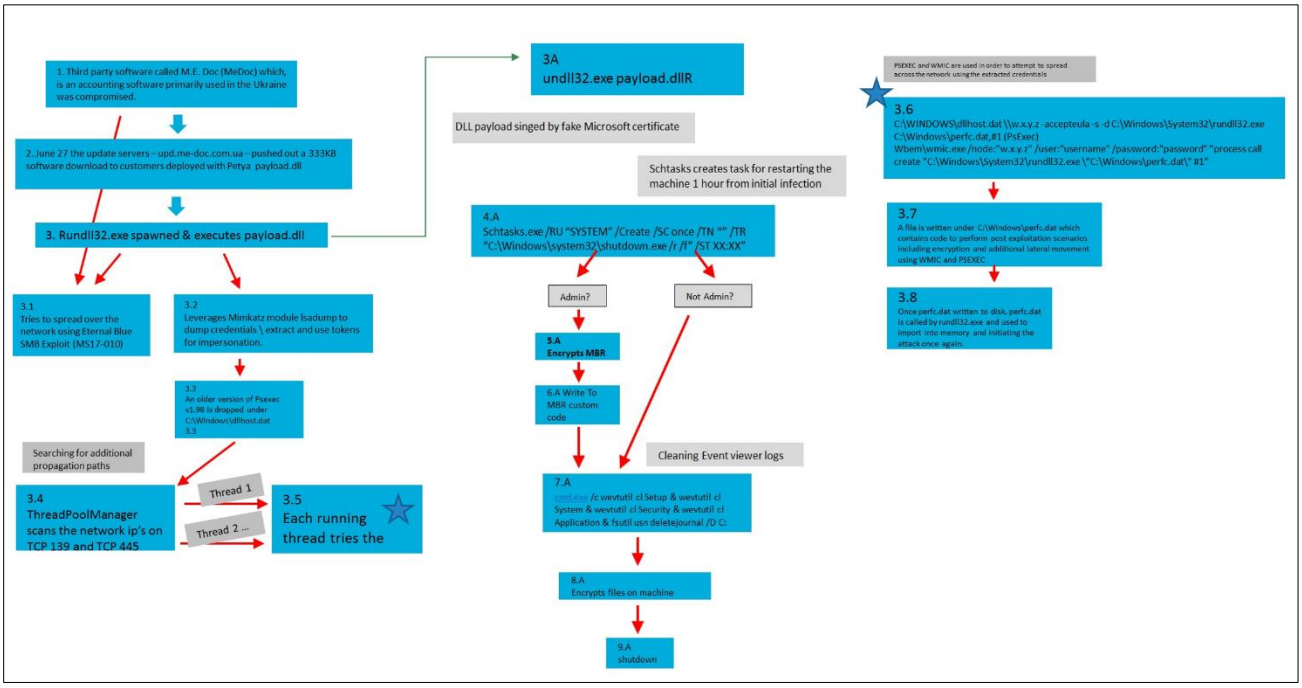

Figure 1. NotPetya Flow Chart, modified from Cynet's NotPetya Technical Analysis

NotPetya Mindmap

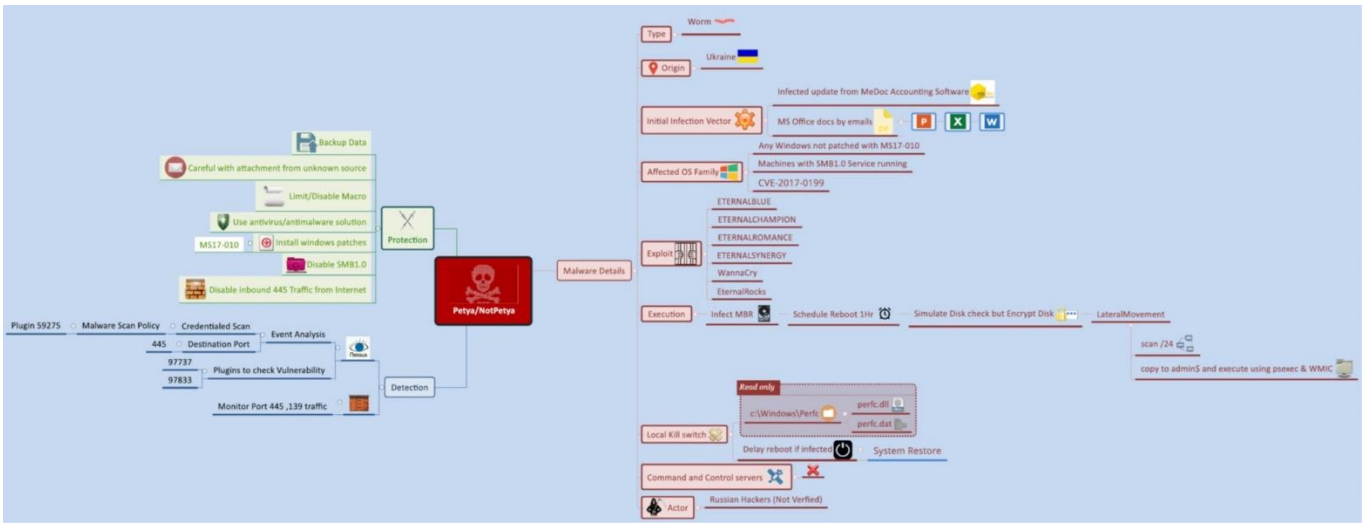

Figure 2. Mindmap of Notpetya, noted for the common practices of mitigation and prevention ransomware malware.