Detecting and Preventing Advanced Persistent Threats

Literature Review

College of Engineering, Mathematics and Physical Sciences

I certify that all material in this dissertation which is not my own work has been identified.

Lewis Lloyd
Student ID No: 680009805

Contents

1 Abstract

Advanced Persistent Threats (APTs) are stealthy, continuous and sophisticated attacks, typically launched against large organisations. This paper examines the current literature regarding Advanced Persistent Threats and determines the trends and patterns between the publications. The paper takes notes from 30 publications surrounding APTs and reveals that our technical analyses have allowed us to fully understand high-profile APTs such as Stuxnet and Duqu. However, the concern lies with the uncharted territory of new attack vectors, such as cloud computing and wireless networks. It seems that the most effective solution is launch quantitative research into the impact of these vectors in order to find what attackers may look to exploit. However, arguments can be made to further the research into machine-learning and AI methods of mitigating APTs.

2 Introduction

As modern technology has evolved, so has the world of industry has. However, one of the disadvantages of integrating technology with industry lies in cybersecurity. The advent of malicious agents targeting industrial systems is widespread and continues to grow [1].

While governments have become targeted by malware, they have also become aware of its power. Attacks such as Stuxnet have caused more damage than military action would have [2]. New threats are actively being funded by governments, and this malware is being developed at an unprecedented level of complexity. The malware studied by academia has clearly had huge investments.

The Advanced Persistent Threat (APT) is a class of malware where threat actors first gain unauthorised access to a network. From this point, they will proceed to remain undetected for long periods of time. The capability for this continuous, undetected attack is what makes APTs so powerful. The malware will typically enter the network through social engineering attacks, and proceed to utilise zero-day privilege escalation exploits in order to execute arbitrary code.

Historically, these attacks have been used for state-sponsored espionage and sabotage. Analysis into the threats shoe that they required extensive test environment to develop. The end result has been malicious gain through damage to economies, environments and public safety [3]. This tells us that governments are confident in their viability as sophisticated threats.

Clearly, the APT is the result of a rapid evolution and investment into malware development. As governments put increasing amounts of resources into these technologies, it is becoming more important than ever to mitigate their effects. As IT professionals, we need to take an active role to prevent spread in the name of privacy, integrity and public safety.

This paper attempts to review current literature surrounding APTs by identifying the key concepts studied within the technical analyses. From this point, it will discuss the trends and patterns within attack vectors, methods of detection, and methods of protection.

We can then look to compare the similarities and differences between publications, and ultimately identify gaps in knowledge. We can then propose where future work should take place in order to be most effective.

3 Advanced Persistent Threats

3.1 Technical Analyses

Cybersecurity is a game of cat-and-mouse. Attackers are constantly finding exploits, and defenders must keep up with them in order to secure their system [4]. Our best chance of detecting and preventing APTs is through analysing past threats. Once we understand how these threats worked, we can start implementing feasible security solutions.

Therefore, in order to develop solutions for the detection and prevention of APTs, it is essential to study past threats in-depth. We need to facilitate the dynamic analysis of these threats in behavioral experiments [5]. This has so far been achieved through executing and monitoring samples of the malware inside controlled environments.

The most famous case is arguably Stuxnet. This APT was found to selectively target a specific Industrial Control System (ICS) by utilising four separate zero-day exploits [6]. It likely required stealing design plans and creating an entire replica test environment.

The malware has been rigorously analysed and was shown to possess incredibly complex features, such as the ability to update itself via its own peer-to-peer network. It also possesses the ability to hide its executable binaries via rootkits [7]. These rootkits exist on both the Windows operating system, as well as within the Programmable Logic Controllers (PLCs) of the infrastructure.

Similar systemic analyses have been conducted into other high-profile cases. It's important to note the similarities between these programs:

- Duqu was a threat likely created by the same team as Stuxnet [8]. This APT sought to steal information instead of sabotage the system. It gained access through exploiting a zero-day exploit in Microsoft Word [9]. It was later discovered that Word's protected view would have prohibited the attack. Therefore, the employee's software was likely not up-to-date.
- Flame was also believed to be from the same team [10]. This malware exploited an MD5 collision in order to register itself as a Windows Update proxy server [11]. It was then able to trick computers on local networks to download its own files instead of Windows Update files.
- Red October looked to utilise exploits in both Microsoft Word and Excel. These exploits allowed attackers to steal credentials, as well as personal information for security questions, and executed these attacks on an international scale in order to steal from multiple governments [12].

The common trend here is creating highly sophisticated malware, yet proceeding to penetrate a system through simple social engineering. More than likely, the initial distributor attempted to trick authorised users into running the zero-day.

The initial reports on these threats have been compiled, analysed and progressed in order to identify common characteristics [2]. Common themes are taking advantage of relaxed internet rules, relying on encryption to prevent signature-based detection of antivirus solutions, and exploiting digital signatures so that malicious binaries could run without trouble.

The ability to evade antivirus detection via XOR encryption is common across all of the malware, which is an efficient way to prevent signature-detection and reverse engineering [13]. XOR is not effective for actual encryption purposes, but appears to circumvent antivirus solutions incredibly well. Additionally, antivirus software often fully skips signed binaries for performance reasons.

Furthermore, APTs can be broken down into three steps; the penetration stage, the propagation stage, and the payload stage [14]. Clearly, the goal must be to either prevent the penetration, or detect and mitigate the propagation, before the payload is activated.

3.2 Attack Vectors

Although gaining unauthorised access to a network can be achieved via directly penetrating the system, a more effective method is through tricking an authorised member of the system into letting the threat actor inside, as members tend to be the most vulnerable endpoint [15].

On the flip side, there exists the hazard of compromised personnel that may have been bribed or blackmailed into opening an attack vector for a malicious agent [16]. This is a factor that does not seem to have been picked up upon by those studying APTs such as Stuxnet, who believe that the initial attack came from the irresponsible use of an infected USB.

Email remains an extremely common communication channel, and as a result of this, malware is typically distributed through a form of social engineering known as spearphishing [17], where emails are targeted towards specific businesses, or even individuals, to increase the chances of success [18].

Especially in more recent times, cloud computing offers a new form of attack vector. As cloud solutions are designed to be accessed from anywhere on the internet, special precautions have to be taken as isolating the network is no longer an option. Cloud storage is an especially difficult vector; since commercial cloud interfaces are designed to hide information from the client, attackers can look to download files over time and produce what will look like standard metrics to avoid being noticed [19].

Similar to how cloud solutions remain open to all access, the advent of wireless networks has led to large numbers internet-enabled devices such as smartphones constantly reading data from any and all sources at any time. Successful attacks have been demonstrated in these environments [20].

3.3 Threat Prevention

Protecting against the initial attack comes down to the key factor of protecting endpoints.

Firstly, administrators should seek to secure network-enabled system hardware. Security solutions have been designed that allow networks to reject traffic if it cannot be decoded. This can prevent the propagation of malware that has encrypted its source before distribution.

Secondly, administrators should look to educate authorised personnel on how to mitigate social engineering attacks. Security training remains a vital part of the software development lifecycle (SDLC), but other fields lack the same security awareness as programmers who understand how attacks can take place at a much deeper level.

Education can be facilitated either through specialised training or through implementing behaviour specifications [21]. However, research appears to indicate that security awareness itself is actually a rare trait among IT professionals [22], and we can assume that there will always be weak-links in systems for as long as people have authorisation to access them. The optimal control approach aims to model an attacker-defender model so that defence strategies can be quantified and compared in order to produce the most effective strategy [23].

Sophos asserts that there is no fully-effectively mitigation against APTs, but attempts to architect protection by wrapping each layer of the client's system with a layer of security [24]. They discovered that most managers aren't aware of what an APT is. If security awareness is lacking at the top of the hierarchy, then most employees are likely lacking in awareness as well. Therefore, safeguarding every stage along the application control path is a good idea.

3.4 Threat Detection

At a certain point, industry professionals respect the fact that securing a system through prevention is ineffective. It only takes a single mistake out of thousands of personnel to compromise a system. The exploit may be something that could never have been anticipated in training or behaviour specifications.

Therefore, the bulk of research takes place in detecting APT attacks, rather than preventing them. Theoretically, it would be better if we were able to catch APT attacks before any harm was captured.

We must seek to detect APT activity as early as possible. Several approaches have been published, including the use of machine learning, monitoring data packets and the statistical analysis of network traffic [3].

One of the more modern advances in research uses Machine-Learning (ML) correlation analysis. Several methods of threat detection have been created, and furthermore, validated with real traffic [25].

- This ML system is compromised of eight separate modules that each aim to detect the early stages of an APT.
- Once a module is triggered, a prediction is made for the progress of the APT. This prediction is based on the probability that it develops into a full APT attack is calculated.
- If there is a high probability, a security team will manually go through steps to mitigate the attack before it completes.

Advancements with machine-learning have significantly outperformed current APT detection systems. These current systems include scanning emails for common words in spearphishing emails [26], identifying data leakages [27] and scanning files for known malicious signatures [28].

An interesting publication is a patent that aims to protect network devices at a low level. The solution authenticates low-level routines at the memory level [29]. In the case that an inauthentic memory command is read, the device isolates itself from the network in order to mitigate propogation.

One final method is through analysing and handling network traffic. Distinguishing healthy traffic from malicious traffic within a system is difficult. When it comes to high-volume traffic, the amount of false positives will require an entire team of security analysts to monitor it.

Current research is being done into identifying the leaner and weaker signals created by APT attacks in order to lower the data bandwidth. It's possible to analyse this data incredibly quickly, and create a ranking of the most suspicious signals. Analysts can then target a small set of signals. This has been shown to be feasible for about 10,000 hosts [30].

4 Conclusion

The in-depth technical analysis of high-profile APTs has allowed us to understand the architecture and functionality of the threats. Both the low-level and high-level mechanisms of these threats. Through this, we have devised successful countermeasures for combating APTs on the platforms where we have already seen them. Furthermore, progress in the detection of APTs is increasing dramatically. The use of machine-learning has facilitated huge spikes in the speed of detection. There are large bodies of research to explore in order to find new techniques.

Experts definitely disagree in the field with qualitative work. With some experts believing in imposing behaviour specifications while others propose more layers of automatic security, it's hard to tell which way to go. A useful study in the future could compare the effects of more imposed security versus more training. One thing we can be certain on is that personnel remain the weakest link within a secure network.

The most significant field lies in the advent of new attack vectors. These have facilitated the need for new countermeasures. Current literature has opted to focus on demonstrating attacks and concepts. As a result of this, the literature lacks quantitative research into the potential impact on new attack vectors.

Further work could seek to properly quantify the impact on enterprise across particular hardware platforms, such as mobile or cloud. Once we understand the areas of risk, we can look to develop methods of prevention before any new malware spreads through the wild. Alternatively, we could look to further the work into machine-learning methods of detecting APT attacks. Machine-learning is an emerging field that could benefit from research into both APT and general methods. It appears to be the most promising path towards the rapid detection of APTs.

References

[1] Aitor Couce Vieira, Siv Hilde Houmb, and David Rios Insua. "A Graphical Adversarial Risk Analysis Model for Oil and Gas Drilling Cybersecurity". *arXiv preprint arXiv:1404.1989*, 2014.

[2] Nikos Virvilis, Dimitris Gritzalis, and Theodoros Apostolopoulos. "Trusted Computing vs. Advanced Persistent Threats: Can a defender win this game?" In: *2013 IEEE 10th International Conference on Ubiquitous Intelligence and Computing and 2013 IEEE 10th International Conference on Autonomic and Trusted Computing*. IEEE. 2013. Pp. 396–403.

[3] Alberto Redondo-Hernández, Aitor Couce-Vieira, and Siv Hilde Houmb."Detection of Advanced Persistent Threats Using System and Attack Intelligence".

[4] David Patten. "The evolution to fileless malware". *Infosecwriters*, 2017.

[5] Christian Rossow et al. "Prudent practices for designing malware experiments: Status quo and outlook". In: *2012 IEEE Symposium on Security and Privacy*. IEEE. 2012. Pp. 65–79.

[6] Thomas M Chen and Saeed Abu-Nimeh. "Lessons from stuxnet". *Computer* 44, pp. 91–93, 2011.

[7] Nicolas Falliere, Liam O Murchu, and Eric Chien. "W32. stuxnet dossier". *White paper, Symantec Corp., Security Response* 5, p. 29, 2011.

[8] Boldizsár Bencsáth et al. "Duqu: Analysis, detection, and lessons learned". In: *ACM European Workshop on System Security (EuroSec)*. Vol. 2012. 2012.

[9] Eric Chien, Liam OMurchu, and Nicolas Falliere. "W32.Duqu: the precursor to the next stuxnet". In: *5th {USENIX} Workshop on Large-Scale Exploits and Emergent Threats ({LEET} 12)*. 2012.

[10] Boldizsár Bencsáth et al. "The cousins of stuxnet: Duqu, flame, and gauss". *Future Internet* 4, pp. 971–1003, 2012.

[11] Alex Sotirov. "Analyzing the MD5 collision in Flame". *Presentation at SummerCon, slides available at http://www. trailofbits. com/resources/flame-md5. pdf*, 2012.

[12] Kaspersky Labs. ""Red October" Diplomatic Cyber Attacks Investigation". *Securelist.com*, 2013. URL: `https://securelist.com/red-october-diplomatic-cyber-attacks-investigation/36740/`.

[13] Carolina Zarate et al. "Analysis of the Use of XOR as an Obfuscation Technique in a Real Data Corpus". In: *IFIP International Conference on Digital Forensics*. Springer. 2014. Pp. 117–132.

[14] Quanyan Zhu and Stefan Rass. "On multi-phase and multi-stage game-theoretic modeling of advanced persistent threats". *IEEE Access* 6, pp. 13958–13971, 2018.

[15] Katharina Krombholz et al. "Advanced social engineering attacks". *Journal of Information Security and applications* 22, pp. 113–122, 2015.

[16] Pengfei Hu et al. "Dynamic defense strategy against advanced persistent threat with insiders". In: *2015 IEEE Conference on Computer Communications (INFOCOM)*. IEEE. 2015. Pp. 747–755.

[17] Grant Ho et al. "Detecting credential spearphishing in enterprise settings". In: *26th {USENIX} Security Symposium ({USENIX} Security 17)*. 2017. Pp. 469–485.

[18] Bimal Parmar. "Protecting against spear-phishing". *Computer Fraud & Security* 2012, pp. 8–11, 2012.

[19] Liang Xiao et al. "Cloud storage defense against advanced persistent threats: A prospect theoretic study". *IEEE Journal on Selected Areas in Communications* 35, pp. 534–544, 2017.

[20] Roger Piqueras Jover and Paul Giura. "How vulnerabilities in wireless networks can enable Advanced Persistent Threats". *International Journal on Information Technology* 1, pp. 145–151, 2013.

[21] Nachaat AbdElatif Mohamed, Aman Jantan, and Oludare Isaac Abiodun. "An improved behaviour specification to stop advanced persistent threat on governments and organizations network". In: *proceedings of the International MultiConference of Engineers and Computer Scientists*. Vol. 1. 2018.

[22] Fadi A Aloul. "The need for effective information security awareness". *Journal of Advances in Information Technology* 3, pp. 176–183, 2012.

[23] Pengdeng Li et al. "Defending against the advanced persistent threat: An optimal control approach". *Security and Communication Networks* 2018, 2018.

[24] Barbara Hudson. "Advanced persistent threats: Detection, protection and prevention". *Sophos Ltd., US February*, 2014.

[25] Ibrahim Ghafir et al. "Detection of advanced persistent threat using machine-learning correlation analysis". *Future Generation Computer Systems* 89, pp. 349–359, 2018.

[26] J Vijaya Chandra, Narasimham Challa, and Sai Kiran Pasupuleti. "A practical approach to E-mail spam filters to protect data from advanced persistent threat". In: *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*. IEEE. 2016. Pp. 1–5.

[27] Johan Sigholm and Martin Bang. "Towards offensive cyber counterintelligence: Adopting a target-centric view on advanced persistent threats". In: *2013 European Intelligence and Security Informatics Conference*. IEEE. 2013. Pp. 166–171.

[28] Nir Nissim et al. "Detection of malicious PDF files and directions for enhancements: A state-of-the art survey". *Computers & Security* 48, pp. 246–266, 2015.

[29] Robert Michael Hussey and Kai J Figwer. "Method and system to protect software-based network-connected devices from advanced persistent threat".

[30] Mirco Marchetti et al. "Analysis of high volumes of network traffic for advanced persistent threat detection". *Computer Networks* 109, pp. 127–141, 2016.