State-of-the-art Attacks on Industrial Control Systems and Their Countermeasures

Kristofor Bolton
University of Wolverhampton,
Wolverhampton Cyber Research
Institude (WCRI)

***Abstract*— State-of-the-art attacks conducted by nation states, or Advanced Persistent Threats (APT), are of increasing concern to multiple stakeholders as attacks and consequences play out in the real world. Attacks in Iran and Saudi Arabia have caused damage to physical systems and threatened the lives of workers within the industrial setting. This paper discusses technical details and intrusion methodologies of state-sponsored attacks on Industrial Control Systems (ICS) and presents countermeasures.**

Keywords—industrial control system, ICS, advanced persistent threat, APT, Stuxnet, Triton

I. Introduction

Advanced Persistent Threat (APT) refers to a threat actor which through significant resources and unusually high technical skill carries out a cyberattack. Undisclosed exploits may be used, so-called zero-day vulnerabilities, and the attacks may last for years [1,4]. Thus, APTs are typically, nation-states, state-sponsored groups or well-financed criminals [3,4]. They perform cyber operations against a wide-variety of industries with the objective to surveil, steal, damage or disrupt and the aim to gain an economic advantage through these actions [2,5].

This paper will analyse two APT attacks and their methods; *Stuxnet*, widely accepted first publicly known APT malware which targeted industrial control systems (ICS) with the aim of physically damaging hardware [7,8], and *Triton*, APT malware which was observed as recently as April 2019 attempting to carry out a similar goal to Stuxnet [10,11]. This paper focuses on the hacking methods and the countermeasures of these attacks.

This paper is organised into five sections: II. Methodology, discussing how this paper forms an unbiased analysis of the topics. III Analysis of Attacks is split into two parts covering the two malwares, Stuxnet and Triton. Section IV Countermeasures discusses the countermeasures to the attack vectors discussed within section III and its associated Appendix D and C which detail each attack vector. Section V discusses future work and acknowledges the limitations of this paper. Finally, conclusions are drawn within section VI.

II. Methodology

This paper aims to present an unbiased analysis of state-of-the-art attacks on Industrial Control Systems (ICS) and countermeasures to these attacks following research guidelines outlined within Machi and McEvoy research methods [6]. The methodology, including inclusion and exclusion criteria, source and search methodology are discussed in Appendix A.

III. Analysis of Attacks

A. *Stuxnet*

The discovery of Stuxnet was the first-time thinking of surrounding confidentially, integrity and accessibility (CIA) which is applied to malware did not fully apply. The aim of Stuxnet was to destroy – physically – a military target. Stuxnet did not need an internet connection, it infected systems via USB drives and targeted real-time devices, such as pumps and valves, and was designed to manipulate controller data which caused the devices to operate outside of their intended thresholds and cause physical damage [7,8].

Chen, et al [7] and Al-Rabiaah [8], agree Stuxnet's differences to common malware as seen in figure 1 taken from Chen, et al's work.

Table 1. Stuxnet's novel characteristics.

Aspect	Stuxnet	Common malware
Targeting	Extremely selective	Indiscriminate
Type of target	Industrial control systems	Computers
Size	500 Kbytes	Less than 1 Mbyte
Probable initial infection vector	Removable flash drive	Internet and other networks
Exploits	Four zero-days	Possibly one zero-day

Figure 1. Stuxnet novel characteristics, Chen et al [7].

Stuxnet spread through the network from the infected USB to infect Windows PCs. It did not, however, necessarily exploit those machines [7,9]. Langer [9] states Stuxnet used fingerprinting to ensure it was on target before activating its payload. This fingerprinting included checking model numbers, configuration details and downloading code from the controller to ensure it was the right controller.

Once a machine matched these target attributes, Stuxnet was activated. Multiple 0-day (B1-3 detailed overview in Appendix B) exploits where used within Stuxnet. The resources required to find unique exploits

are not insignificant, supporting the suggestion Stuxnet was created by an APT.

In addition to 0-days, Stuxnet also utilised other novel methods; valid digitally signed certificates from Verisign [12]. These valid certificates allowed Stuxnet to be executed each time an infected system was booted, without raising suspicion [12]. This evidence suggests somewhat significant resourcing behind the design of Stuxnet, as these certificates would have been stolen and were not used anywhere else – thus were not likely purchased on the Dark Web.

B. Triton

The discovery of the Triton attack framework marked the first time ICS malware was discovered to affect industrial safety systems [10,11], and thus directly threaten human life. To date, only two instances of infection have been observed, both within the Middle East [11].

Triton represents a different methodology compared to Stuxnet. Through their methods, it is clear the authors of Stuxnet had access to intimate knowledge of the facilities they targeted [9], however, the authors of Triton clearly have limited knowledge. They were forced to reverse engineer a proprietary protocol, TriStation, to infect ICS safety systems [10,11]. FireEye [11] and Wetezels, et al [10] agree Triton uses custom and commodity tools to achieve its aims, these are detailed within Appendix C.

The Triton Framework allowed attackers to load attack scripts and run automated attacks [16]. According to FireEye's analysis the attackers switched to custom tools when attempting to avoid anti-virus or when entering a critical phase within the attack, such as before taking over engineering workstations [16]. This was likely to reduce the likelihood of failure and have maximum control over their tools.

Attackers gained access into the system via a spoofed Triconex log reviewing application (Triconex is the product name of the ICS Safety controllers). The payload consisted of two binary files; inject.bin and imain.bin, which allowed the reverse engineered protocol TriStation to be used to reprogram the controller [10,11]. Wetezels et al, suggest basic anti-forensic measures were used, as if the reprogramming failed the TRITON framework would write a dummy program to memory [10].

Triton differs from Stuxnet in it does not implement 0-day exploits, however, equivalency can be derived from their shared use of highly customised attack tools [7,10].

Figure 2 shows details of the attack, including the tools used to conduct the attacks:

TOOL	COMPONENTS	PURPOSE	ATTACK LIFECYCLE STAGE: Initial Compromise	Establish Foothold	Escalate Privileges	Internal Reconnaissance	Move Laterally	Maintain Presence	Complete Mission
SecHack	KB77846376.exe	Credential harvesting			x	x			
	KB77846376.exe.x64								
NetExec	NetExec.exe	Remote command execution					x		
	runsvc.exe	NetExec runner							
Cryptcat-based backdoor	cryptcat.exe cryptsvc.exe svchostpla.exe	Backdoor							
	compattelprerunner.exe	C&C domain name generator		x					
	ProgramDataUpdater.xml	Scheduled task file (persistence mechanism)							
PLINK-based backdoor	napupdatedb.exe	Backdoor		x				x	
Bitvise-based backdoor	alg.exe userinit.exe csrss.exe	Backdoor							
	tquery.dll txflog.dll cryptopp.dll DEFAULT DEFAULT.BAK	Backdoor components					x	x	
OpenSSH-based backdoor	spl32.exe WinSAT.exe csrss.exe	Backdoor							
	clusapi.dll PolicMan.dll verifier2.dll misc.mof setup.ini	Backdoor components					x	x	
WebShell	logoff.aspx	Modified legitimate Outlook Web Access Component							
	flogon.js	Modified legitimate Outlook Web Access Component				x		x	
	ftpexts.tlb	Output file containing credentials harvested by logoff.aspx							

FIREEYE

Figure 2 Commodity Intrusion Tools table, FireEye [11]

IV. COUNTERMEASURES

Internationally recognised standards provide a starting point for all businesses. Standards such as the ISO 27001 series, NIST Cybersecurity Framework cover a broad number of controls which if implemented will reduce the threat from insiders, data exfiltration and network attacks among others [13,15]. Stuxnet reportedly used insider actors [7,8,9], more robust controls, such as those within ISO 27001 may have prevented this.

Countering APTs is a difficult challenge, they are well resourced groups of individuals [3,4]. Targeting this aspect, Defence-in-Depth (DiD) provides a layered approach to security, separating network entities to both provide technical and economic disincentive, and prevent or slow an attacker's progress [14]. DiD may prevent or increase the steps required for 0-days and rootkits to be used by attackers, as seen with Stuxnet and Triton (Appendix B and C respectively).

However, DiD cannot be relied upon alone. Its main deterrence is to increase the time and thus cost to an attacker [14]. Triton shows how persistent APTs are - the attackers reverse engineered an undocumented protocol and created working malware. Active detection

methods, such as data mining and network analysis should be combined with DiD.

Langer [9] points out code signing could have prevented the major vulnerability Stuxnet exploited. The ICS controllers which Stuxnet targeted treated syntactically correct code as legitimate, no matter the source. Digital code signing for verification ensures data comes from a trusted source. Additionally, this creates an extra step for APTs to traverse, increasing cost for the attacker.

When dealing with such advanced and persistent attackers, it would be prudent to consider your systems already compromised. This attitude may result in greater attention and stopping attacks before they reach their final stages. While it may sound pessimistic, consider such attacks last on average 78 days [5]. It is realistic to consider victims can discover an attack within that time if they are actively looking. 'Threat hunting' cyber security businesses, such as Cybereason, take this approach with end-point protection and data mining services designed to discover ongoing attacks [17].

V. Future Work and Limitations

Future work based on this topic can improve and build on this paper by surveying an increased number of APT attacks on ICS. For example, Flame, Duqu, Shamoon and Industroyer are not covered within this survey.

This research draws heavily from technical analysis conducted and released by commercial cyber security companies. Due to the fact this necessary information is not peer-reviewed, it is cross referenced with other cyber security companies research to verify the claims and findings. Of course, this research also draws heavily on peer-reviewed work, however, it must also be noted that this work too is based off work by the cyber security companies who discovered the malware.

VI. Conclusion

This paper has successfully answered the questions is sought to answer described within Appendix A.

The research and analysis of the Stuxnet and Triton APT malware reveals the use heavy use of commodity attacks, which are not particularly complex or unique. It is the way in which these techniques are used or combined with the persistence and skill of the APTs which makes them successful. The use of custom exploits, often including multiple previously undiscovered exploits, so-called zero-days, is noteworthy among the attacks analyzed.

Initial attack vectors can vary, these are often dependent on the target. Industries of critical importance should consider cyber security a priority and assume they are a target and act based on standards, such as ISO 27001 and advice given by national governments.

Countermeasures can be affective against APTs. Typically, defense in-depth techniques can result in stronger and more resilient networks and provide a disincentive for adversaries. Additional countermeasures are discussed within section IV.

References

[1] FireEye (2019a) *Advanced Persistent Threat Groups*. [online]. Available: https://www.fireeye.com/current-threats/apt-groups.html

[2] Kaspersky (2019) *What is an Advanced Persistant Threat (APT)?* [online]. Available: https://www.kaspersky.com/resource-center/definitions/advanced-persistent-threats

[3] Cisco (2019) *What is an Advanced Persistent Threat (APT)?* [online]. Available: https://www.cisco.com/c/en/us/products/security/advanced-persistent-threat.html

[4] FireEye (2019b) *APT1 Exposing One of China's Cyber Espionage Units.* [online]. Available: https://www.fireeye.com/content/dam/fireeye-www/services/pdfs/mandiant-apt1-report.pdf

[5] FireEye (2019c) *M-Trends 2019*. [online]. Available: https://content.fireeye.com/m-trends

[6] Machi. L. A., and McEvoy. B. T. (2009) *The Literature Review*. Crown Press, Thousand Oaks, CA, USA.

[7] Chen, T., and Abu-Nimeh, S. (2011) Lessons from Stuxnet. In IEEE *Computer*. 2011. Vol. 44. Issue. 4. Pp. 91-93. [online]. Available: https://ieeexplore.ieee.org/document/5742014

[8] Al-Rabiaah, S. (2018) The "Stuxnet" Virus of 2010 As an Example of A "APT" and Its "Recent" Variances. In *2018 21st Saudi Computer Society National Computer Conference (NCC)*. 25-26 April 2018. Riyadh, Saudi Arabia. [online]. Available: https://ieeexplore.ieee.org/document/8593143

[9] Langer, R. (2011) Stuxnet: Dissecting a Cyberwarefare Weapon. In *IEEE Security & Privacy*. Vol. 9, Issue 3. May-June 2011. Pp. 49-51. [online]. Available: https://ieeexplore.ieee.org/document/5772960

[10] Wetezels, J., and Meijer, C. (2018). *Analysing the TRITON industrial malware*. [online]. Available: https://www.midnightbluelabs.com/blog/2018/1/16/analyzing-the-triton-industrial-malware

[11] FireEye (2019d) *TRITON Actor TTP Profile, Custom Attack Tools, Detections, and ATT&CK Mapping.* [online]. Available: https://www.fireeye.com/blog/threat-research/2019/04/triton-actor-ttp-profile-custom-attack-tools-detections.html

[12] Symantec (2011). *W32.Stuxnet Dosier*. [online]. Available: https://www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/w32_stuxnet_dossier.pdf

[13] BSI Group (2013). *ISO/IEC 27001 Informaiton Security Management.* [online]. Available: https://www.bsigroup.com/en-GB/iso-27001-information-security/

[14] SANS (2001) *Defense in Depth*. [online]. Available: https://www.sans.org/reading-room/whitepapers/basics/paper/525

[15] NIST (2019) *NIST Cybersecurity Framework*. [online]. Available at: https://www.nist.gov/cyberframework

[16] FireEye (2017) *Attackers Deploy New ICS Attack Framework "TRITON" and Cause Operational Disruption to Critical Infrastructure*. [online]. Available: https://www.fireeye.com/blog/threat-research/2017/12/attackers-deploy-new-ics-attack-framework-triton.html

[17] Cybereason (2019) *Cybereason*. [online]. Available: https://www.cybereason.com/

APPENDIX A

Appendix A discusses the methodology used to carry out the survey conducted for this paper.

A. Research Questions

What is the state-of-the-art within APT attacks on ICS? What are the attack vectors APTs typically use? What are the methods of attack APTs use? How can the attacks discussed be mitigated or countered?

B. Inclusion and Exclusion Criteria

Research articles with the following criteria appear in this paper:

- Published within the last 5 years, unless fundamental or noteworthy.
- Discuss state-sponsored attacks on ICS environments or countermeasures to such attacks.

Research articles with the following criteria will be excluded from this paper:

- Non-peer reviewed articles.
- Biased articles, such as white papers.
- Not written in English.

Technical analysis articles, written by industry experts or commercial entities with a good reputation will be referenced within this work, specifically: FireEye and Kaspersky.

C. Source and Search Methodology

Data sources specific to computer science and technology research were used: IEEE Xplore, ACM Digital Library, Science Direct and Cornell University's arXiv.org. Logical operators such as 'OR' and 'AND', and other search manipulation operators, such as double quotes, were used to force specific phrase searches, e.g. "APT ICS". Additionally, advanced search features were utilised to limit results to research articles from the last five years, once an initial search without limitations was conducted to find possible fundamental papers.

APPENDIX B

	Stuxnet Attack Vectors	
	Attack Vector	**Description**
B1	Win32k.sys Local Privilage Escalation (MS10-073)	0-day privilage escalation. Vulnerability within function pointer table; index into the table not fully validated allowing arbitary code execution [12].
B2	MS10-061 Remote Execution Vulnerability	0-day remote execution. Vulnerability within Print Spooler allowing files to be wrtten to %System% folder. Stuxnet used this vulnerability to copy and execute installation on remote vulnerable machines [12].
B3	MS08-067 Arbitrary Execution	0-day arbitrary execution within Windows Server SMB service. Sending malformed path string over SMB allowed arbitrary execution. Stuxnet used this to copy itself onto remote systems [12].
B4	MrxNet.sys Windows Rootkit	MrxNet.sys is a rootkit with a compromised legitimate Realtek digital certificate. The rootkit creates a new device object which intercepts IRP requests (writes, reads to NTFS, FAT or CD-ROM) [12].

APPENDIX C

	Triton Attack Vectors	
	Attack Vector	**Description**
C1	Py2EXE	Spoofed Triconex log reviewing application, containing the TRITON Framework [11].
C2	Cryptcat Backdoor	Executes daily using Program DataUpdater and
		NetworkAccessProtection UpdateDB [11].
C3	PLINK Backdoor	Executes daily using Program DataUpdater and NetworkAccessProtection UpdateDB [11].
C4	NetExec.exe	A custom lateral movement and remote execution tool [11].
C5	SecHack.exe	A custom credential harvesting tool [11].