Keywords: advanced persistent threat, cybersecurity, detection

Cybertrust

An Approach for Detection of Advanced Persistent Threat Attacks

Qingtian Zou and Peng Liu, The Pennsylvania State University

Xiaoyan Sun, California State University, Sacramento

Anoop Singhal, National Institute of Standards and Technology

Advanced Persistent Threat (APT) campaigns employ sophisticated strategies and tactics to achieve their attack goal.

The evolution of APT strategies and tactics compounds the challenge of detecting attack campaigns. This article introduces an approach whose purpose is to assist cybersecurity analysts in detecting such attacks.

APT Fundamentals

APTs are one of the top cybersecurity concerns in enterprise networks. In the past decade, APT attack campaigns such as Ghostnet in 2009, Stuxnet in 2010, and Deep Panda in 2015 resulted in large-scale data breaches (e.g., stealthy data exfiltration during weeks or months), system infection, integrity degradation, denial-of-service, and even damage to cyber-physical systems such as centrifuges used for separating nuclear material.[1]

Guided by a carefully designed playbook, an APT attack campaign can be carried out over a long time interval (e.g., several months). An example of an APT strategy is to let two attack actions be attributed to different user accounts, IP addresses, port numbers, and time intervals, especially when there is data or control dependency between them. The commonly used tactics can be APT malware combining probing, infection, backdoors, monitoring, and stealth. Such strategies and tactics typically consist of multiple steps, each playing a different role, such as performing initial access, malicious code execution, privilege escalation, and data exfiltration. In each step the APT campaign will employ *techniques* such as spear phishing, drive by download, buffer overflow, and pass the hash. An *APT tactic* is usually recognized as a chain of specific APT techniques. To be stealthy, APT campaigns usually make their individual attack actions unnoticed by defenders through means such as backdoors and rootkits. Even if the attack actions are noticed by a defender the actions will appear to be random and uncorrelated.

Figure 1 shows an example of a five-step APT tactic[2]. The first three attack actions are against a Domain Controller (DC) in a Microsoft Windows environment, whereas the last two are against another user machine in the Windows domain. This tactic can start with a compromise somewhere in the supply chain, with the adversary inserting malicious software or hardware components into legitimate applications or systems before the systems or applications reach their end-users.[3] For privilege escalation, the attacker bypasses the User Account Control (UAC). The UAC prompts the user for confirmation when a process (i.e., executing program) requests system-level privileges. By bypassing the UAC the attacker can escalate privileges without being noticed. After that the attacker can dump the user credentials such as password hashes. The attacker can then try to use the credentials to access other machines. If the attacker is successful at completing the first four steps then he or she can download data from the target machine.

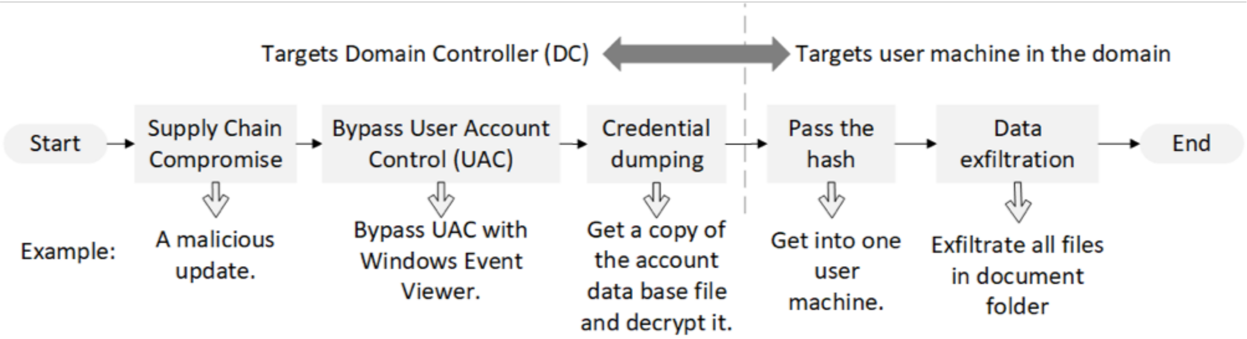
Targets Domain Controller (DC)
Targets user machine in the domain
Start
Supply Chain Compromise
Bypass User Account Control (UAC)
Credential dumping
Pass the hash
Data exfiltration
End
Example:
A malicious update.
Bypass UAC with Windows Event Viewer.
Get a copy of the account data base file and decrypt it.
Get into one user machine.
Exfiltrate all files in document folder

Figure 1. A sample 5-step APT Tactic [1]

Due to the stealthy nature of APTs, APT campaigns are not easy to detect. Although Indicators of Compromise (IoCs) are being regularly provided to security analysts to serve as alerts or cues, for a particular threat an APT's big picture is often not recognized until it is too late to avoid devastating damage. An APT campaign can often hide its individual attack steps as a proverbial "needle in a haystack" of probes from ordinary non-persistent attacks, thus thwarting the security analyst's ability to recognize the causality and dependencies between the APT's individual attack steps. Without "gluing" these individual attack steps together, there is no basis upon which to assess the adversary's intent, specific objectives, and strategy.

Common Characteristics of APT Attack Campaigns

The landscape of an APT attack campaign can be substantially broader than what is shown in Figure 1. The following are common characteristics of APT attack campaigns:

- **Multi-attack-stage**. The campaign usually contains multiple attack stages, each of which consists of multiple steps. Each stage and step within a stage has a specific purpose (e.g., authenticating to a remote server to set up a covert channel for data exfiltration). In addition, the campaigns typically follow a particular attack pattern, that is, a specific sequencing of APT techniques.
- **Control and Data Dependency.** Each APT technique in an APT tactic has its post-conditions and pre-requisites. Post-conditions are the results of an attack action, such as malicious processes being created, files being accessed, or user account being modified.

Pre-requisites describe the preconditions that are needed for the APT technique to be effective. In many cases, the pre-requisite of an APT technique in an attack step is the post-condition of the APT technique in the previous attack step. The APT techniques can be chained to form an APT tactic through matching the post-conditions and pre-requisites.

- **Malware.** To achieve and automate an APT campaign, the adversary has strong incentives to install and run malware that serve different purposes, some even on a server that is hosted by a "bullet-proof" hosting company.
- **Data Exfiltration.** Data exfiltration is concerned with what data are exfiltrated and how. Once desired data has been collected, adversaries often package the data through compression and encryption to avoid detection. The package is usually removed after data exfiltration to eliminate the traces left on the compromised computer.
- **Stealth.** Since an APT campaign may last weeks or even months, an APT campaign often hides its individual attack steps as a "needle" in a "haystack" of probes from ordinary non-persistent attacks. This makes it difficult for the analysts to notice the causality and dependencies between the individual attack steps.
- **Alert Sources.** The attack actions involved in an APT campaign often leave traces of themselves. These traces could be recorded or audited by system or security sensors in a variety of data types, such as the Tcpdump data, system audit logs, and firewall logs, etc. In addition, security analysts (in a cybersecurity operation center) are constantly working on harvesting IoCs and reasoning the relationships between the identified IoCs.
- **Threat Intelligence.** Threat intelligence means collecting out-of-band intelligence about the threat actors, including individuals, groups, or organizations that are believed to be operating maliciously behind a particular APT campaign.
- **Intrusion Set.** As defined in Structured Threat Information Expression (STIX)[4], an intrusion set is "a grouped set of adversarial behaviors and resources with common properties believed to be orchestrated by a single threat actor."
- **Vulnerability.** APT campaigns may exploit a set of vulnerabilities to achieve their goals. Some APT campaigns may exploit zero-day vulnerabilities.
- **Attribution.** Attribution means answering the "who did what" question through associating the attack actions involved in an APT campaign with particular threat actors.

Table 1 contains a review of several past APT campaigns, including their impacts and the associated key aspects.

Table 1. A Review of Several Past APT Campaigns

APT Campaign	Year	Impacts	Key Characteristics
Titan Rain	Since 2003	Titan Rain has caused distrust between several countries. As the first instance of state-sponsored espionage from China that was made public, Titan Rain triggered a decades-long effort by the U.S. government to reduce the breadth and scope of Chinese cyber operations against U.S. targets.	Data exfiltration; Threat intelligent

SkiPot	Since 2006	Intellectual properties, including design, financial, manufacturing and other information are leaked.	Multi-attack-stage; Control and data dependency; Malware; Intrusion set; Vulnerability
GhostNet	2009	GhostNet has infiltrated high-value political, economic and media locations in 103 countries. Computer systems belonging to embassies, foreign ministries and other government offices were compromised.	Stealth; Alert sources; Attribution
Stuxnet	2010	This campaign targets supervisory control and data acquisition systems. Specifically, Stuxnet targets programmable logic controllers (PLCs), which allow the automation of electromechanical processes such as those used to control machinery and industrial processes including gas centrifuges for separating nuclear material. Stuxnet is believed to be responsible for causing substantial damage to the nuclear program of Iran.	Malware; Stealth; Threat intelligent; Intrusion set
Deep Panda	Since 2012	This threat group is known to target many industries, including government, defense, financial, and telecommunications. The intrusion into healthcare company Anthem in 2014 has also been attributed to Deep Panda.	Malware; Stealth; Attribution

Current Approaches to APT Detection

From the foregoing list of characteristics, it is evident that detection of APT tactics is more challenging than that for run of the mill intrusion or malware detection. The classical intrusion detection and malware detection methods have been shown to be inadequate for defending against APT tactics. Signature-based methods can be easily fooled because some APT techniques do not even have distinctive signatures; anomaly detection-based methods may suffer from a high false positive rate; and rule-based methods need extensive manual effort to keep the rules up to date while still suffering from high false positive rates.

Despite the challenges, the cybersecurity community is actively pursuing more sophisticated detection capabilities, some of which rely on bottom-up approaches and others are top-down.

Bottom-up approaches try to infer the existence of APT tactic with low-level information. For example, provenance-tracking (a.k.a., information-flow tracking) is a representative bottom-up approach. Existing APT recognition approaches[5,6,7,8] are primarily based on unsupervised "connecting the dots" through provenance-tracking across multiple events or activities that may seem legitimate individually but signal malice collectively. Without the need to leverage domain-specific knowledge about APTs, these approaches first conduct activity-dependency analysis and causality-graph construction, and then use heuristics to simplify the resulting graphs. Security analysts then examine the graphs for telltale evidence APT campaigns. However, this type of approach has drawbacks such as the dependence-explosion problem that is well known in digital forensics.

Top-down approaches take advantage of knowledge about known APT, skirting for instance the dependence explosion problem: They rely on models of the workings of APTs such as Stuxnet so unsupervised learning of associations is not a necessity. For example, HOLMES uses the APT life cycle as the attack model, and then tries to match every single step.[9] The HOLMES system gathers computer audit data and ranks the severity of APT attacks in real time. An APT attack is classified based on seven stages of the so-called APT "kill chain": 1) perform initial compromise; 2) establish foothold; 3) escalate privileges; 4) conduct internal reconnaissance, 5) Move laterally, 6) maintain presence; and 7) complete mission. HOLMES determines if an APT attack occurred based on the severity of activities in each APT stage. Each stage has a severity score. An APT attack is considered as happened if the weighted sum of these scores is high.

A New Approach for Detecting APT Attacks

We constructed an approach, illustrated in Figure 2, for top-down approaches to detection of APT attacks. The idea behind the approach is to identify the employed APT techniques through data analysis and then match these techniques to a specific APT tactic. The APT repository contains known APT tactics that the techniques can be mapped onto. To start the APT detection, system logs and configuration files are collected and fed into corresponding data parsers. The APT technique identifiers are responsible for detecting the APT techniques used in each individual attack step. These techniques are mapped to the APT tactics by APT matcher. Specifically, the APT matcher will take the parsed APT tactics from the APT parser, match the individual APT techniques to the known APT tactics, and ultimately generate APT instances.

After the matching is complete, the APT ranker will rank all the APT instances based on the completeness and output a ranking list. Although the approach may seem to be bottom-up due to the component of matching lower layer APT techniques to APT tactics, it is actually top-down. The APT repository already contains known tactics. It will only identify individual APT techniques that are present in those known APT tactics.

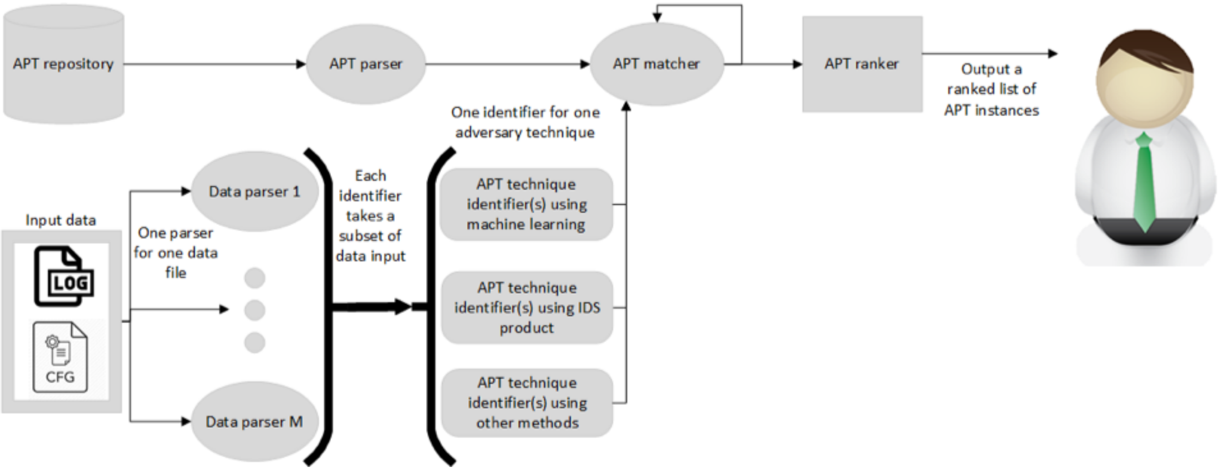
APT repository
APT parser
APT matcher
APT ranker
Output a ranked list of APT instances
One identifier for one adversary technique
Input data
LOG
CFG
One parser for one data file
Data parser 1
Data parser M
Each identifier takes a subset of data input
APT technique identifier(s) using machine learning
APT technique identifier(s) using IDS product
APT technique identifier(s) using other methods

Figure 2. Our APT Detection Approach [1]

Each phase in the approach generates its own analysis results. The result of a later phase is based upon the findings from the previous phase. New findings of a previous phase will trigger another round of analysis in the next phase. For example, the analysis outcome of the technique identification phase is the identified technique and its post-conditions. At the end of the technique-identification phase, if a new technique is identified, the tactic matcher should be automatically triggered. Similarly, the analysis results of the tactic matching phase are the new or updated APT tactic instances. At the end of the tactic matching phase, once a new APT tactic instance is created or the previous instance is updated, the tactic ranking phase should be automatically triggered. The tactic ranker then updates APT tactic instances' completeness and ranking results accordingly. The ranking of an instance is based on the number of steps completed in that instance.

The variety and sophistication of cyber attacks that include APT attacks are increasing at a global level. The monetary loss caused by APT attacks can be huge. There is an urgent need to develop an approach for fast detection of such attacks. In this article we have briefly described an approach for detection of APT attacks. To validate the proposed approach, we have implemented a proof-of-concept system prototype, and evaluated how useful it is by running experiments to detect a set of seven APT tactics, including the one shown in Figure 1. We found that its effectiveness is mainly influenced by the APT Repository that holds precise and comprehensive knowledge, whether the individual APT technique identifiers can raise alerts in a timely manner, and the coupling between different yet concurrently happening APT campaigns. In addition, we observe that the knowledge (about the dependencies between the attack actions in a tactic) held in the APT Repository inherently makes the approach more resilient to false alarms raised by individual APT technique identifiers.

Disclaimer

The views and conclusions contained herein are those of the authors and should not be interpreted as representing the policies or endorsements, either expressed or implied, of their employers. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes, notwithstanding any copyright annotations thereon. Certain commercial entities, equipment, or materials may be identified in this document to describe an experimental procedure or concept adequately. Such identification is not intended to imply recommendation or endorsement by the authors or their employers, nor is it intended to imply that the entities, materials, or equipment are necessarily the best available for the purpose.

References

1. D. Kushner, "The real story of Stuxnet," *IEEE Spectrum*, Feb. 26, 2013. Available: https://spectrum.ieee.org/telecom/security/the-real-story-of-stuxnet.
2. Q. Zou, A. Singhal, X. Sun, and P. Liu, "Automatic recognition of advanced persistent threat tactics for enterprise security," in *Proc. Sixth Int. Workshop on Security and Privacy Analytics*, 2020, pp. 43-52.
3. Supply Chain Compromise. The MITRE Corporation, McLean, Va. Available: https://attack.mitre.org/techniques/T1195/

4. Structured Threat Information Expression. Cyber Threat Intelligence Technical Committee, OASIS Open Consortium, Burlington, Mass. Available: https://oasis-open.github.io/cti-documentation/
5. H. Lee, X. Zhang, and D. Xu, "High accuracy attack provenance via binary-based execution partition," in *Proc. Network and Distributed Syst. Security Symp.*, 2013. Available: https://www.ndss-symposium.org/wp-content/uploads/2017/09/03_1_0.pdf.
6. S. Ma, X. Zhang, and D. Xu, "Protracer: Towards practical provenance tracing by alternating between logging and tainting," in *Proc. Network and Distributed Syst. Security Symp.*, 2016. Available: https://friends.cs.purdue.edu/pubs/NDSS16.pdf.
7. S. Ma, J. Zhai, F. Wang, K. H. Lee, X. Zhang, and D. Xu, "MPI: Multiple Perspective Attack Investigation with semantic aware execution partitioning," in *Proc. 26th USENIX Security Symp.*, 2017, pp. 1111-1128. Available: https://www.usenix.org/system/files/conference/usenixsecurity17/sec17-ma.pdf.
8. M. Hossain, S. Sheikhi, and R. Sekar, "Combating dependence explosion in forensic analysis using alternative tag propagation semantics," in *Proc. IEEE Symp. on Security and Privacy*, 2020, pp. 1139-1155.
9. S. Milajerdi, R. Gjomemo, B. Eshete, R. Sekar, and V. Venkatakrishnan, "HOLMES: Real-time APT detection through correlation of suspicious information flows," in *Proc. IEEE Symp. on Security and Privacy*, 2019, pp. 1137-1152. doi: 10.1109/SP.2019.00026.

Vita: Qingtian Zou *is doctoral student in Informatics at Pennsylvania State University, University Park. Contact him at qzz32@psu.edu.*

Vita: Xiaoyan Sun *is an Assistant Professor with California State University, Sacramento. Contact her at xiaoyan.sun@csus.edu.*

Vita: Peng Liu *is the Raymond G. Tronzo, MD Professor of Cybersecurity and founding Director of the Center for Cyber-Security, Information Privacy, and Trust at Pennsylvania State University. Contact him at pliu@ist.psu.edu.*

Vita: Anoop Singhal *is a Senior Computer Scientist and Program Manager in the Computer Security Division at the National Institute of Standards and Technology. Contact him at psinghal@nist.gov.*