The 12th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications
7-9 September, 2023, Dortmund, Germany

APT Detection: an Incremental Correlation Approach

Salva Daneshgadeh Çakmakçı[1] , Georgios Gkoktsis [2] , Robin Buchta[3] ,
Kai Oliver Detken[1] , Felix Heine[3] , Carsten Kleiner[3]
[1] DECOIT GmbH & Co. KG, Bremen, Germany, daneshgadeh@decoit.de, detken@decoit.de, www.decoit.de
[2] Fraunhofer SIT — ATHENE, Darmstadt, Germany, george.gkoktsis@sit.fraunhofer.de, sit.fraunhofer.de
[3] Hochschule Hannover - University of Applied Sciences and Arts, Hanover, Germany,
robin.buchta@hs-hannover.de, felix.heine@hs-hannover.de, carsten.kleiner@hs-hannover.de, hs-hannover.de

***Abstract* — Advanced Persistent Threats (APTs) are a growing and increasingly prevalent threat. Current detection systems focus primarily on individual procedures and create alerts on this foundation. To effectively detect APT attacks, which rarely consist of single activities, individual alerts must be correlated to comprehensively encapsulate APT activity and provide better situational awareness to the operators. We use this to initiate targeted and proactive countermeasures and thus improve overall security. This paper presents a correlation engine that uses alarms from standard rule-based systems and correlates them with each other. We evaluate the proposed solution using an APT scenario as an example and discuss the advantages and disadvantages of this approach. We argue that the fast, simple implementation, which is an add-on to SIEM, must be considered when evaluating the limited informative value of rule-based systems in the face of zero-day exploits or even sophisticated living-off-the-land attacks.**

Keywords* — *Advanced Persistent Threat (APT), Rule-based System, SIEM, Correlation Engine, APT Detection, Cyberattack Detection

I. Introduction

Advanced Persistent Threats (APTs) [1] are a growing concern for organizations of all sizes, as they pose a significant risk to their operations and reputation. The motivation of an attacker can be difficult to interpret, and organizations cannot rely on the assumption that they will not be targeted. Hence, all organizations must do their best to protect themselves from potential threats to ensure the continuity of their operations.

However, small and medium-sized enterprises (SMEs) face unique challenges in achieving effective cyber defense, as they may lack the necessary resources and the management motivation to invest in such solutions. Managed service solutions offer a practical solution to this issue by utilizing commercial off-the-shelf (COTS) data in a generalizable manner, making the cost of protection more predictable and manageable. This is also the area of existing solutions, such as Security Information Event Management (SIEM) systems, which specialize in conventional attacks.

The current state of the APT, as drawn from recent threat reports from the cybersecurity community [2]–[7], shows an evolution of the cybercrime ecosystem towards an industrialized space. This means that an end-to-end cyberattack may not originate from the same threat actor, but rather from multiple, as is the example of Ransomware-as-a-Service. Specialized groups

sell their services online, or even offer subscription based services. As such, detecting APT activity transcends stopping a single adversary poised on inflicting damage.

Rule-based systems, in particular, are well suited for protection against APTs because they extract indicators of compromise (IoC) from previously generated Cyber Threat Intelligence (CTI), enabling a more targeted defense against specific attacks that are often a subset of APT end-to-end malicious activity. Unlike more advanced methods, such as artificial intelligence-based solutions, rule-based systems mostly do not rely on the normal behavior of a target area as a basis for threat detection. Despite their general volatility, their field of applicability is wide and cross organizational, since there is no retraining requirement, as is the case with AI-based methods. Furthermore, AI-based methods are in the developmental stage concerning APT detection and are not yet suitable for operational use. Two cutting-edge AI-based Advanced Persistent Threat (APT) detection systems, namely ThreaTrace [8] and ShadeWatcher [9], have emerged from research efforts, demonstrating remarkable technical advancements. However, despite their technical breakthroughs, these systems remain unsuitable for operational deployment due to their current state as scientific prototypes.

The approach we propose is based on alarm correlation. Alarms are generated from a singular, or different rule-based systems [1] and correlated with each other via a correlation engine so that serious incidents can be considered in a prioritized manner. The research questions for this study

[1]In the context of this paper, the term *System* refers to the definitions of Intrusion Detection Systems. In this context, different detection systems mean the deployment of a variety of tools that on their own merit are considered individual IDS.

979-8-3503-5805-6/23/$31.00 ©2023 IEEE

are: i) How can existing rule-based systems be adapted to the requirements of APT attacks by correlating events? and ii) What is the trade-off of using rule-based systems in terms of APT detection?

The main contribution of the work is the investigation of existing rule-based methods in terms of the necessary adaptations to meet the increased requirements for APT attack detection. The solution approach, the hierarchical correlation of different rule-based detection engines, is evaluated practically, using an exemplary APT scenario. Furthermore, we discuss the advantages and disadvantages of the implementation and give an outlook for the further development in view of the new threat situation.

The remaining structure of the paper is as follows: Section II presents the related work in this line of research and argues our differentiating novelty. Section III details our methodological approach in detail, the technology and infrastructure that is used, the analytical methods, the threat model, and the exemplary attack scenario. Section IV presents the selected ruleset and attack platform and finaly, Section VI contains our concluding remarks.

II. Related Work

There are different approaches in the field of APT attack detection, which we briefly present here. On one hand, there are approaches that try to pin APT attacks down to a specific point of the attack. Exemplary works for this approach are references [10]–[12]. [10] tries to detect APTs at the DNS level, [11] with malware detection and [12] by identifying command and control (C&C) communication.

Another category that inherently maps correlations in the data structure are graph-based approaches, which can have different characteristics. For instance, Sleuth [13] uses a rule set to map the normal behavior of the system from provenance graphs to graph data and detects deviations accordingly, or threaTrace [8], which detects anomalies through Graph Neural Networks. In addition to provenance graphs, knowledge graphs can also be used to associate assets with events in the system. Methodologically closer to our approach are hierarchically arranged rule systems, such as [14] and [15]. Both systems use a hierarchical concept of rules to bring insights from log information to a cyber-kill-chain representation.

In addition to hierarchical control systems, there is another class of systems that correlate existing alarms. These include, among others, [16]–[18]. NoDoze [16] aims to reduce false positive alarms by generating dependency graphs from existing alarms. However, this approach is based on graph data to generate the dependency graphs afterwards, while we create correlations based on alerts only, with no intermediate stages. Reference [17] works on the alert correlation of IP addresses. The authors use a multi-stage Markov property process. They use only the IP address to create correlations.

The closest to our work that we could identify is [18], where an approach similar to ours with different detection modules and a subsequent correlation of the individual events is taken. They use a Hidden Markov Model, whereas we rely on a simpler form of feature correlation.

III. Methodology

We developed a detection method and then established a virtual environment to test our proposed approach. The kill-chain model was adapted to understand the characteristics and behaviors of an attack and design a detection framework. Operating system audit logs and network traffic were collected and used by the detection framework. Three levels of abstractions were defined for the creation of rules as follows:

1) Detection rules for matching an event (network traffic or endpoint event) with pre-defined conditions.
2) Correlation rules for correlating low-level rules which may hold the potential to evolve into an APT alarm.

A. Framework

This section includes details of our proposed approach. Figure 1 demonstrates the high-level architectural design of the employed SIEM for the purpose of APT detection.

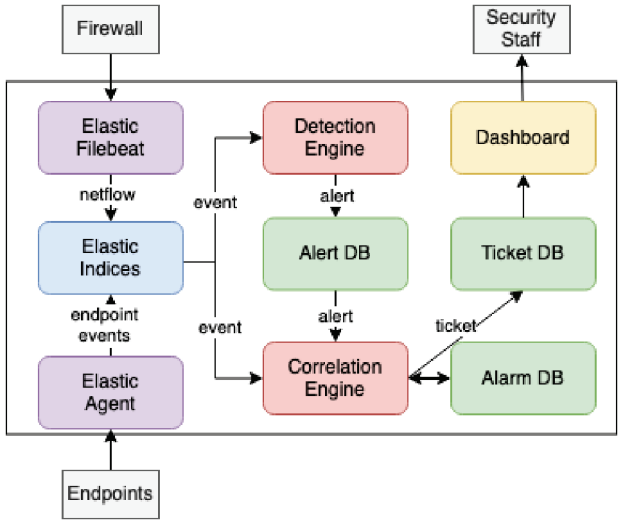

Figure 1. High-Level Architectural Design of the SIEM

Our proposed framework is embedded in a research-based SIEM. The Elastic stack (Elasticsearch, Logstash, Kibana, ELK) [2] is the backend of our SIEM for raw data storage, visualization and parsing. The detection engine is a component for the execution of detection rules and the creation of alerts. Low-level alerts are stored in the Alert DB, a database for that specific purpose. The data providers of detection rules are Elastic Agent

[2]https://www.elastic.co/elastic-stack/

[3] and Elastic Filebeat[4]. Elastic Agent collects metrics, traces, availability, security, network, and application data from each host and ingests them to Elasticsearch. Filebeat is a Lightweight shipper for forwarding data logs. In our proposed framework, it is installed on a Firewall (Opensense) and ships Netflow [5] data to Elasticsearch. It supports monitoring of whole network traffic.

Finally, the correlation engine correlates alerts utilizing correlation rules and storing them within the Alarm DB. The second responsibility of the Correlation Engine is to generate Tickets when the cumulative probability of correlated alerts exceeds the threshold value. A ticket includes a chain of alerts, along with the underlying reasons behind the alerts.

B. Rules and Alerts

Each rule consists of condition(s), a schedule, a risk score and tag(s). When conditions are met, alerts will be created. A Rule condition always consists of a query (simple or aggregation). For aggregation queries (e.g., number of reached ports based on distinct source and destination IP addresses), a threshold value also can be set. Therefore, a rule first executes the query and then compares the result to the threshold value. A schedule defines a time interval between each run of a rule. There are four different levels of tags in our system as follows:

1) MITRE ATT&CK tactics and techniques tags: Table I demonstrates seven potential stages of APT with their corresponding MITRE tactic identification numbers, as defined in [19].
2) Protocol and application tag: It defines an application or protocol used as an attack vector.
3) Asset identifier tag: It provides information about the asset (e.g., IP address, hostname) to which technical resilience measures should be applied.
4) Affected asset tag: It provides asset information to the correlation.

An alert is a JSON object which includes information about the rule conditions, which caused the match and triggered an alert, such as source/destination IP/Port number, the total number of sent/received bytes or cardinally of a data attribute. An alert also keeps information about timestamp, risk scores and associated tags.

C. Correlation Engine

The correlation engine incorporates correlation rules that assess whether the rule conditions are collectively met by the low-level alerts and alarms within predefined time periods. Since APT attacks usually evolve in seven kill-chain stages (namely, Initial Access, Reconnaissance, C&c, Privilege Escalation, Lateral Movement, Execution and Impact), it is almost impossible to detect an APT attack in real, or near-real time. We hypothesize that the incremental correlation approach can fit well with the gradually evolving structure of APT attacks. As depicted in figure 1, the correlation engine periodically fetches alarms from Alarm DB and correlates them with low-level alerts originating from individual hosts. The correlation engine continuously integrates alerts into alarms and finally creates tickets for those alarms that have surpassed the APT attack confidence value (a probability that an alert might be an APT attack) threshold. The ticket should include comprehensive information about each associated low-level alert within ticket to make the attack traceable by security experts. To achieve this objective, the Correlation Engine is also responsible to correlate alerts with raw events to extract detailed information about the underlying reason for alerts.

D. Threat Model and Attack Scenario

We model our adversary as an APT attacker, who also exhibits living-of-the-land (LOL) behavior. Table I summarizes the attack stages that we consider:

Table I. ATTACK STAGE OF THE END-TO-END ATTACK

Attack stage & ***Description***	
Initial Access (tag: TA0001)	The attacker gains an initial foothold in the target network.
Reconnaissance (tag: TA0043)	The attacker explores their target environment and identifies assets, network topology and system information from OSINT sources before initial access.
Command and Control (tag: TA0011)	A communication channel between the compromised infrastructure and the attacker's infrastructure is established.
Discovery (tag: TA0007)	Once inside the target environment, the attacker attempts to explore and identify further targets, vulnerabilities, and valuable assets.
Privilege Escalation and Persistence (tag: TA0004, TA0003)	The attacker attempts to elevate their access privileges by gaining root/administrator privileges, or other.
Lateral Movement (tag: TA0008)	The attacker moves inside the target network. Both north-south and east-west movement is considered.
Credential Access (tag: TA0006)	The attacker attempts to steal credentials from valid accounts to enable Persistence, Privilege Escalation, or Lateral Movement.
Exfiltration and Impact (tag:TA0043, TA0040)	The attacker proceeds to accomplish their objective by either exfiltrating the gathered information or producing a malicious effect.

After analyzing multiple AttackIQ [20] posts on realistic adversary emulation of actual past use cases, we constructed an attack scenario against a hypothetical company with the following stages:

1) A spearfishing campaign is launched targeting specific users of the company,
   As a result, an initial foothold is established.
2) The deployed payload checks in with a pre-existing C2 infrastructure and maintains a constant encrypted channel, using HTTPS.

[3]https://www.elastic.co/elastic-agent

[4]https://www.elastic.co/beats/filebeat

[5]https://www.cisco.com/c/en/us/products/ios-nx-os-software/ios-netflow/index.html

3) Once checked in, the attacker manually enters LOL commands to explore the blue infrastructure, map the network and discover any useful data for exfiltration.
4) Additionally, to files and data, the attacker attempts to steal valid account credentials, by using tools, such as Mimikatz to dump credentials from the memory of the infected host.
5) At this stage, the Discovery phase is completed and the attacker attempts to move laterally to the protected segment. They use a variety of techniques to achieve that, such as exploiting WMI or SMB or directly connecting through SSH.
6) Once the DC is accessed, they search for valuable data for exfiltration, alter the DC Access policies.
7) Once access to the objective target is achieved, they attempt to discover databases and manually change their entries for impact. Additionally, an attempt to interfere with the Domain is made, by virtue of altering valid users' passwords or creating new Domain users.

IV. Experimental Methodology

In this section, we explore the detection rules that can be employed to identify various stages within the presented attack scenario III-D and the correlation rules which were used to create a ticket with an APT . To accommodate the limited space in this paper, we have created a GitHub repository [6] where the exact query of the detection rules can be found. We approached the attack stages through three distinct methods. Firstly, we devised rules capable of precisely detecting the attack stage based on its unique signature. Secondly, we developed rules that focus solely on tracking the signature of well-known adversaries' groups. As a result, only attack techniques that have been previously modelled into the MITRE ATT&CK model can be effectively detected. Lastly, the final group of attack stages present significant challenges in terms of detection, either due to their inherent complexity or the high likelihood of generating false positives. Stages 3 and 4 can be readily detected through rule-based mechanisms. On the other hand, stages 1, 2, and 7 fall within the second group and the remaining 6th stages falls into the third group. Since the rules in the first group are straightforward to comprehend, we will focus our discussion in this paper on the remaining rules from the second and third groups.

Considering attack simulation for the experiments, we opt for the MITRE CALDERA framework [21], as it fulfills the following requirements: i) it is a red teaming emulator, which is a sufficiently granular approximation of real APT activity, and ii) it offers a transparent and repeatable experimentation methodology, as only the publicly available abilities are considered.

[6]https://github.com/GeorgeGkoktsis/APT-Detection-An-incremental-correlation-Approach-Rule-Repository

a) Spearphishing Attachment (T1566.001): Detecting malicious Microsoft (MS) Word files through spearphishing is a difficult task that requires monitoring multiple events on hosts. One approach is the implementation of a rule that specifically identifies suspicious child processes associated with MS Office applications such as Word, PowerPoint, and Excel.

b) Application Layer Protocol: Web Protocol (T1071.001): Detecting C&C communication through application layer protocols has significant challenges. On this basis, well-known patterns utilized by prominent threat actors, such as IP addresses, port numbers and hash values play a significant role in identifying malicious traffic. For example, the FIN7 threat group commonly utilized registered domains which follow a pattern such as `[a-zA-Z]4,5.(pw|us|club|info|site|top)$`[7]. Additionally, to facilitate the detection of C&C channels, a rule can be written to compare network observations against indicators from CTI feeds.

c) Valid Accounts (T1078): Using existing credentials of valid accounts is a common strategy employed by threat actors to camouflage their malicious activities and successfully evade detection. Rule-based detection methods are not well-suited to differentiate the usage of valid account credentials by either the legitimate user or an adversary. In such scenarios, user behavioral anomaly detection algorithms demonstrate better performance than rule-based systems. They model normal user behavior and catch deviations from normality, such as abnormal login times, the use of unusual commands, or unexpected remote system access. As a result, we do not rely on rules to diagnose the usage of valid accounts to evade detection.

d) Lateral Movement (TA0008) Related Techniques (T1021.00X): Remote access protocols such as RDP, SSH, SMB, and WMI are widely used among legitimate users throughout the enterprise network. Therefore, individual rules which track each remote access type most probably result in an overwhelming number of alerts (increased false positives rate). Therefore, as discussed in the previous paragraph (IV-0c), behavioral analyses have demonstrated better results in detecting malicious activities via remote access protocols. However, user activities such as the creation or modification of Windows executable files over network shares, alterations to OpenSSH binaries, enabling remote SSH login via system setup, installation and usage of specialized applications like SharpRDP, and the use of specific Windows built-in utilities like PsExec could be considered as strong clues for lateral movement attempts. [8].

e) Account Manipulation (T1098): Generating entirely new privileged accounts and modifying credentials

[7]https://www.mandiant.com/resources/blog/fin7-pursuing-an-enigmatic-and-evasive-global-criminal-operation

[8]For more information, visit: https://github.com/elastic/detection-rules/tree/20d2e92cfe0cc0b473a88c4c21c0e6fd306cce85/rules

of existing accounts are commonly employed approaches by adversaries to ensure continuous access to victim systems. In our designed attack scenario, the adversary creates a user and assigns it to a privileged group to maintain persistent control over the victim system.

f) Account Manipulation (T1098): Some adversaries threaten the availability and integrity of systems. For example, erasing or altering raw data at rest can threaten the availability and integrity of the entire system. WhiskeyAlfa and StoneDrill [9]are examples of such wiper malware. As Wiper malware usually need to get raw access to the hard disk, the presence of read/write access to the Master Boot Record or Disk Volume Partition could be an indicator of Wiper malware.

A. Correlated Alerts

Here, we present two correlation rules designed to correlate low-level alerts generated by detection rules and low-level alerts and alarms generated by correlation rules. These correlation rules are based on three key dimensions: spatial, temporal, and methodical (MITRE techniques and tactics).

- Spatial Dimension: refers to the correlation of various low-level alerts at different hosts, or otherwise based on entity telemetry data, such as network IPs, or user account metadata.
- Methodical Dimension: refers to the correlation of different low-level alerts with different MITRE tactic and technique IDs.
- Temporal Dimension: refers to the correlation of low-level alerts within a suitable time window.

For example, if two low-level alerts each associated with different MITRE ATT&CK tactics (Methodical) are generated for the same source IP address (Spatial) during the last 24 hours (Temporal) a correlation engine creates an alert for that IP address using the correlation rule.

The optimization of these three dimensions is beyond the scope of this paper because our main concern is designing the correlation engine and developing exemplary correlation rules. However, It should be emphasized that a longer time window can fit better to capture stealthy APT attacks and a narrower time window may result in higher false positive, but more timely alerts. APT attacks such as Stuxnet [22] and Flame [23] were discovered after several years of consistently being active. Therefore, it's important to monitor events continuously, and adjust time windows corresponding to the target environment (e.g., based on event frequency and criticality of their systems).

We implemented two correlation rules in a three-dimensional correlation space. The first correlation rule creates a precautionary APT alarm when at least three different detection rules have generated low-level alerts, which as a whole have at least 3 distinct MITRE Technique tags for a single host within a relatively short time window (we selected 30 minutes as the initial value). This rule aims to identify potentially suspicious activity that may indicate an ongoing APT attack on a specific host. The second correlation rule has two steps. First, it checks if the Alarm DB includes any alarm from the last 30 days. If this condition is met, it examines if a minimum of three distinct detection rules have generated low-level alerts, which as a whole have at least 3 distinct MITRE Tactic tags for at least two different hosts within the last 24 hours. Alternatively, if the Alarm DB includes any alarm from the last 30 days, the correlation rule continues to correlate alarms with low-level alerts while maintaining the same set of conditions. Each time that the correlation engine creates an alarm based on the second correlation rule, the probability of APT alarm will increase. The precise calculation of this probability, if at all plausible, is also beyond the scope of this paper. For future work we will improve the number and features of correlation rules, for example, a correlation rule should check if there is a semantic relationship between different MITRE Tactics which are diagnosed on different hosts.

It is expected that the reference to the collection of events as an APT alarm in this paper is intended solely to emphasize its significance and does not necessarily imply that it is an APT.

V. Discussion

This section discusses the limitations and opportunities of rule-based systems for APT attack detection. As mentioned in Section II, rule-based systems can have different variations. The limitations and opportunities listed here refer to the approach discussed in this paper, which is based primarily on CTI and the correlation of individual events at different dimensions.

One of the key advantages of rule-based systems is that community-driven or commercial rule sets can be used, as a significant portion of CTI is publicly available, allowing many providers and rule creators to focus on the same problems, especially when standard systems are in use. This allows everyone to work with nearly the same data and threat basis, making result exchange possible.

Furthermore, these systems are often easily extendable by adding new rules to the existing rule set. Additionally, the rules focus on the attacks, which are generally much less frequent than misclassification of normal behavior, thus reducing the chance of false positive alarms and increasing alarm fidelity. This requires proper rule creation and leads to the next advantage, where faulty rules can be easily identified by tracing the generated alarm back to the underlying rules, allowing targeted adjustments. The rules can also take any level of granularity, reflecting the natural form of CTI, as rules can target specific indicators, and a hierarchical structure is possible, leading to an implicit correlation of events.

The high accuracy and fidelity alarms generated in this manner substantially reduce the cognitive load of

[9]https://attack.mitre.org/techniques/T1561/001/

responders, a critical bottleneck in Incident Response. Apart from that, they are naturally compatible with Threat Hunting and can be used as inputs to more advanced, AI-powered analytics.

On the other hand, the disadvantages become particularly apparent when the properties of APTs are considered, as rules based on CTI can only detect known patterns. As mentioned in section IV, rules are sometimes more useful depending on the technique under consideration than for others, where more behavior-based methods are better. Due to the determination of APTs and their ability to adapt to the target's defenses, rule-based systems are more of an annoyance to the attacker but not a hindrance. Despite this deterrent functionality, i.e., the APT attacker may also reconsider their motivation and potentially abandon secondary targets[10], they are ultimately ineffective against a belligerent attacker poised on maintaining stealth.

VI. Conclusions

This paper presents a rule-based APT attack detection scheme, which correlates atomic intrusion detection alerts to form a high-level APT intrusion alarm. Our detection infrastructure consists of the ELK stack as the basis for data storage, processing and visualization, Elastic Agent and Filebeat for host and network source data collection and an event correlation engine. The Control Validation Compass tool provided the base source for our detection ruleset. Then, informed by the current state of the APT landscape, we constructed an attack scenario, testable with the public abilities of the MITRE CALDERA framework and developed a simple network testbed on Proxmox. As future work, we plan to validate the method and evaluate its performance in our network simulation testbed.

Acknowledgment

The authors would like to thank the German Federal Ministry for Economic Affairs and Climate Action (BMWK) for the financial support, as well as all other partners involved in the research project SecDER[11] for their great collaborations.

References

[1] J. T. F. T. Initiative *et al.*, *SP 800-39. managing information security risk: Organization, mission, and information system view*. National Institute of Standards & Technology, 2011.

[2] S. X-Ops, *Sophos 2023 Threat Report Maturing criminal marketplaces present new challenges to defenders*. Sophos, 2022.

[3] Microsoft, *Microsoft Digital Defense Report 2022*. Microsoft Corp., 2022.

[4] T. Micro, *Defending the Expanding Attack Surface Trend Micro 2022 Midyear Cybersecurity Report*. Trend Micro Research, 2022.

[5] Dragos, *ICS/OT CYBERSECURITY YEAR IN REVIEW 2021*. Dragos Inc., 2022.

[6] ENISA, *ENISA Threat Landscape 2022*. European Union Agency for Cybersecurity, ENISA, 2022.

[7] Crowdstrike, *2022 Global Threat Report*. CrowdStrike, 2022.

[8] S. Wang, Z. Wang, T. Zhou, H. Sun, X. Yin, D. Han, H. Zhang, X. Shi, and J. Yang, "THREATRACE: Detecting and Tracing Host-Based Threats in Node Level Through Provenance Graph Learning," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 3972–3987, 2022.

[9] J. Zengy, X. Wang, J. Liu, Y. Chen, Z. Liang, T.-S. Chua, and Z. L. Chua, "SHADEWATCHER: Recommendation-guided Cyber Threat Analysis using System Audit Records," in *2022 IEEE Symposium on Security and Privacy (SP)*, 2022, pp. 489–506.

[10] M. S. Manggalanny and K. Ramli, "Combination of DNS traffic analysis: A design to enhance APT detection," in *2017 3rd International Conference on Science and Technology - Computer (ICST)*, 2017, pp. 171–175.

[11] G. Laurenza, R. Lazzeretti, and L. Mazzotti, "Malware Triage for Early Identification of Advanced Persistent Threat Activities," *Digital Threats*, vol. 1, no. 3, aug 2020.

[12] A. Alageel and S. Maffeis, "Hawk-Eye: Holistic Detection of APT Command and Control Domains," in *Proceedings of the 36th Annual ACM Symposium on Applied Computing*, ser. SAC '21. New York, NY, USA: Association for Computing Machinery, 2021, p. 1664–1673.

[13] M. N. Hossain, S. M. Milajerdi, J. Wang, B. Eshete, R. Gjomemo, R. Sekar, S. D. Stoller, and V. N. Venkatakrishnan, "SLEUTH: Real-Time Attack Scenario Reconstruction from COTS Audit Data," in *Proceedings of the 26th USENIX Conference on Security Symposium*, ser. SEC'17. USA: USENIX Association, 2017, p. 487–504.

[14] S. Wen, N. He, and H. Yan, "Detecting and Predicting APT Based on the Study of Cyber Kill Chain with Hierarchical Knowledge Reasoning," in *Proceedings of the 2017 VI International Conference on Network, Communication and Computing*, ser. ICNCC 2017. New York, NY, USA: Association for Computing Machinery, 2017, p. 115–119.

[15] S. M. Milajerdi, R. Gjomemo, B. Eshete, R. Sekar, and V. Venkatakrishnan, "HOLMES: Real-Time APT Detection through Correlation of Suspicious Information Flows," in *2019 IEEE Symposium on Security and Privacy (SP)*, 2019, pp. 1137–1152.

[16] W. U. Hassan, S. Guo, D. Li, Z. Chen, K. Jee, Z. Li, and A. Bates, "NoDoze: Combatting Threat Alert Fatigue with Automated Provenance Triage," in *26th Annual Network and Distributed System Security Symposium, NDSS 2019, San Diego, California, USA, February 24-27, 2019*. The Internet Society, 2019.

[17] F. Xuewei, W. Dongxia, H. Minhuan, and S. Xiaoxia, "An Approach of Discovering Causal Knowledge for Alert Correlating Based on Data Mining," in *2014 IEEE 12th International Conference on Dependable, Autonomic and Secure Computing*, 2014, pp. 57–62.

[18] I. Ghafir, K. G. Kyriakopoulos, S. Lambotharan, F. J. Aparicio-Navarro, B. Assadhan, H. Binsalleeh, and D. M. Diab, "Hidden Markov Models and Alert Correlations for the Prediction of Advanced Persistent Threats," *IEEE Access*, vol. 7, pp. 99 508–99 520, 2019.

[19] B. E. Strom, A. Applebaum, D. P. Miller, K. C. Nickels, A. G. Pennington, and C. B. Thomas, "Mitre att&ck: Design and philosophy," in *Technical report*. The MITRE Corporation, 2018.

[20] (2022) Attackiq. [Online]. Available: https://www.attackiq.com/

[21] A. Applebaum, D. Miller, B. Strom, C. Korban, and R. Wolf, "Intelligent, automated red team emulation," in *Proceedings of the 32nd Annual Conference on Computer Security Applications*, 2016, pp. 363–373.

[22] A. Matrosov, E. Rodionov, D. Harley, and J. Malcho, "Stuxnet under the microscope revision 1.1. eset. 2010," 2011.

[23] B. Bencsáth, G. Pék, L. Buttyán, and M. Félegyházi, "skywiper (aka flame aka flamer): A complex malware for targeted attacks," *CrySyS Lab Technical Report, No. CTR-2012-05-31*, 2012.

[10] The term "secondary targets" refers to targets of lesser importance to the APT attacker than the primary target but are still part of the overall attack strategy

[11] https://secder-project.de