IEICE TRANSACTIONS

on Information and Systems

DOI:10.1587/transinf.2024EDL8046

Publicized:2024/08/20

This advance publication article will be replaced by the finalized version after proofreading.

A PUBLICATION OF THE INFORMATION AND SYSTEMS SOCIETY

The Institute of Electronics, Information and Communication Engineers

Kikai-Shinko-Kaikan Bldg., 5-8, Shibakoen 3 chome, Minato-ku, TOKYO, 105-0011 JAPAN

LETTER

Practical APT Group Hash Unit Profiling Framework Using TTPs

Sena LEE[†], **Chaeyoung KIM**[†], ***and*** **Hoorin PARK**[†,††],

SUMMARY With the rise of cyber threats, identifying APT groups becomes increasingly crucial for enterprise security experts. This paper introduces a comprehensive framework for profiling APT groups, focusing on Lazarus and APT29. It underscores the critical role of malware hash unit profiling in contemporary cyber security efforts, aiming to fortify organizational defenses against evolving APT threats.

key words: *APT Group Profiling, Cyber Threat Intelligence, APT29, Lazarus*

1. Introduction

Advanced Persistent Threat (APT) involves continuous and covert hacking techniques primarily targeting governments and corporations. To counter such threats, many organizations invest significant time and effort in APT tracking.

Despite research efforts to counter these cyber-attacks, the infrastructure for large-scale events like the Tokyo Olympics faced 450 million attempts[1]. It has been challenging to achieve meaningful results with conventional APT tracking systems [2]. This difficulty is compounded by the increasing number of APT groups and the evolution of its tactics. Although it provides comprehensive statistical results, it does not show correlations between attack techniques through in-depth analysis rather than just listing statistics [3].

Accordingly, this study provides various analysis of the four factors to show how to derive meaningful content. Previous studies proposed a quantitative scoring framework revealing APT cyber attacks usually have higher scores than fileless attacks [4]. In this context, this study suggests severity scoring rather than quantative scoring of APT cyber attacks using the integration of Tactics, Techniques, and Procedures (TTPs) datasets with the Common Attack Pattern Enumeration and Classification (CAPEC) attack pattern especially Typical severity.

This paper proposes a framework that conducts various statistics and analyzes each malware hash unit, which indicates attack activities. We use the keword 'unit' to represent statistics per a single hash. It introduces a practical framework assigning score to each malware hash unit based on tactics and techniques.

Transitioning from an operation-based to a hash-based profiling framework allows precise measurement of the risk levels of APT behaviors. This shift enhances risk assessment accuracy and allows for a more systematic approach to security threats. The proposed framework, built on public datasets and verified for performance, can be utilized by various organizations. This framework can help organizations better understand and mitigate the risks associated with APTs by providing a more granular and comprehensive risk assessment.

[†]The authors are with the Department of Information Security, Seoul Women's University.

[††]Corresponding author: Hoorin Park (e-mail: hrpark@swu.ac.kr).

2. Proposed Framework

The proposed framework comprises three main components: **Malware Hash**, **Automation System**, and **Analysis System**.

2.1 Overview

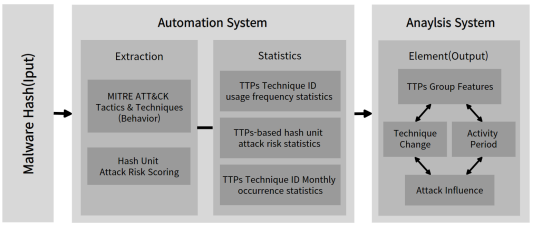

Fig. 1 Model structure of the proposed framework.

Fig.1 illustrates our system model, which is divided into an automation system and an analysis system. The analysis system examines three key elements related to the analysis of data specifically extracted from an open site, VirusTotal [5]: i) **TTPs Group Features** based on the usage frequency statistics of Technique IDs' monthly occurrence statistics. ii) **Technique Change** derived from Technique ID monthly occurrence statistics. iii) **Activity Period and Attack Influence** is analyzed using TTPs-based hash unit attack risk statistics.

2.2 Malware Hash

To achieve information related to Lazarus, we extracted malware hashes from 607 issues occured in 2023 on Lazarus.day [6], a community-based website that shares information about Lazarus. This site stores hash values discovered in relation to attacks. As for APT29, hash values were extracted from the Apt malware dataset on GitHub [7].

2.3 Automation System

The automation system comprises two stages: **Extraction** and **Statistics**, facilitated by code utilizing publicly available data. The collection of data and statistics are each performed in the 'Extraction' and 'Statistics' stages.

2.3.1 Extraction

Extraction 1: MITRE ATT&CK Tactics and Techniques

The first type of data extracted is MITRE ATT&CK Tactics and Techniques, available from Virustotal. If the malware hash value in VirusTotal is input, the information of MITRE ATT&CK Tactics and Techniques is shown on the 'BEHAVIOR' page. This dataset was collected by web-crawling.

Extraction 2: Malware Hash Unit Attack Risk Scoring

The threat level of the malware hash is measured automatically by using a scoring algorithm associated with TTPs.

$$\text{score}_{\text{technique}} = w_{\text{tactic}} \times \frac{\sum_i \text{score}_i}{n} \tag{1}$$

$$\text{score}_{\text{hash}} = \sum_i \left(\text{score}_{\text{technique},i}\right) \tag{2}$$

Briefly explained, the algorithm flow measures the threat score for a single hash. Among the several techniques, a single technique score is calculated by extracting scores about 6 elements. This technique score is extracted by Eq. 1, and the sum of all techniques is the risk score for the attack hash. Then the sum is calculated Eq. 2. These processes are described in Algorithm 1.

The hash input in the threat score calculation algorithm extracts TTPs through VirusTotal. In Eq. 1, 'score_i' refers to the score for each element and 'w_{tactic}' is the weight value of the tactic for technique. 'n' is the number of the scores. Additionally, the six scoring elements are as follows: Tactic, Required Permission, Effective Permission, Supports, Remote, Defense Bypassed, and Procedure Examples. The six scoring method and weight value is described in Cho et al., 'An Apt Attack Scoring Method Using MITRE ATT&CK' [8]where this is tabulated in Table 14 and Table 15.

Algorithm 1 Calculate Threat Score Hash

```
procedure CALCULATE THREAT SCORE HASH(hash_input)
    hash_score ← 0
    techniques ← extract_techniques(hash_input)
    for technique ∈ techniques do
        technique_score ← w_tactic × (Σ_i score_i) / n
        hash_score ← hash_score + technique_score
    end for
    return hash_score
end procedure
```

Unlike the risk of APT attacks, which are assessed based on operations, this paper aims to determine the overall risk of a group's attack on a hash basis. This method is more useful than existing methods for analyzing the attack flow of APT groups. It also provides a lot of statistical basis for predicting changes in future attack tactics.

2.3.2 Statistics

Statistics 1: Technique ID usage frequency statistics

In the 'Extraction' process, information from MITRE ATT&CK Tactics and Techniques was utilized to conduct statistical analysis. In these 'Statistics', the frequency of usage for each technique assigned to specific tactics is aggregated based on data obtained during 'Extraction'. The collected frequency data is used to analyze the attack patterns of specific groups. According to the extracted dataset, statistics for the top 20 techniques used most frequently were generated for each APT group.

Statistics 2: Technique ID Monthly occurrence statistics

Similar to 'Statistics 1', TTP extraction data is used for statistical analysis. Dates were extracted based on the 'first submission' in virustotal by hash units. With this, the statistics were formed. In other words, statistics were generated on how frequently the Top 20 techniques were used on a monthly basis.

Statistics 3: TTPs-based hash unit attack risk statistics

In the 'Extraction' process, hash unit statistics are created and calculated using Attack Risk Scoring information. These statistics determine monthly trends in the risk scores.

3. Framework performance analysis

Based on the three extracted statistics, the framework performance was assessed through analysis of Lazarus and APT29.

3.1 Analysis System

The analysis was conducted centered around three elements pertaining to three statistics extracted from the automated system. These three elements: TTPs Group Features (3.1.1), Technique Change (3.1.2), and Activity Period with Attack Influence (3.1.3), represent the results extracted through the analysis of the corresponding statistics for each element.

3.1.1 TTPs Group Features

We examine three techniques in Lazarus and APT29. **Lazarus** will be analyzed in the following.

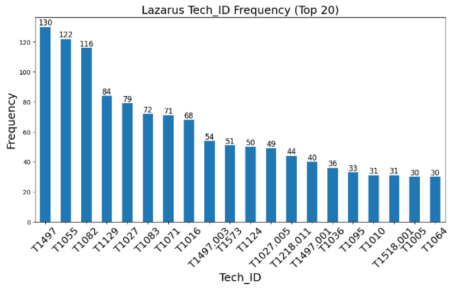

Fig. 2 Frequency of use of the top 20 techniques of Lazarus.

The most frequent Technique ID in Fig.2 is T1497. **T1497** is a technique that detects systems operating in a virtualization. In "AppleJeus Operation", this technique involves waiting a specified amount of time before downloading the second stage payload. And it is used in "Operation Dream Job", using the GetTickCount and GetSystemTimeAsFileTime data collection tools.

The second most frequent **T1055** is used for the injection of malicious code into the memory space of a trusted process. "Operation Dream Job" uses this technique to inject malicious DLL libraries into victims by sending malicious Docx. files containing fake job offers.

The fourth most frequent **T1129** is used or executing shared modules, including arbitrary payloads, on a victim

system. This technique increases the success rate of the attack itself. In “Operation Dream Job”, this technique is used to collect victim information after infection.

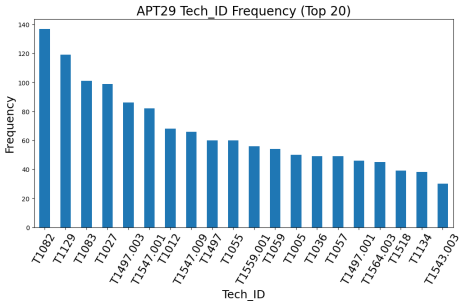

Fig. 3 Frequency of use of the top 20 techniques of Apt29.

Next, we provide a detailed analysis of the attack techniques employed by **APT29**. The most frequent Technique ID in Fig.3 is T1082. **T1082** is used for retrieving system information. It is useful in a cloud environment, where authenticated API calls can return data of the virtual machines. “Operation Ghost” installed the FatDuke backdoor to collect various system information. In the SolarWinds Compromise, APT29 checked available disk space.

The third most frequent **T1083** involves file scanning, allowing an attacker to gather file system data and shape their subsequent actions. It has been used in cases such as “Operation Ghost” and “SolarWinds Compromise” for directory enumeration.

The fourth most frequent **T1027** allows the attackers to encrypt the system in transit, making executable files more difficult to detect. An attacker can exploit command obfuscation to obscure commands executed in a payload. In “Operation Ghost,” APT29 used steganography to hide a payload within an image.

3.1.2 Technique Change

In this subsection, an analysis of attack samples is provided along with the correlation of statistics from the past (2008 through 2015) through the statistics in Fig.4 and Fig.5. Especially, this paper focuses on the sub-techniques that have the highest threat scores based on Algorithm 1. The attack samples are by Lazarus in 2024, and APT29 in 2023 [9], [10]. Sequentially, the technique changes of **Lazarus** and **APT29** will be analyzed over time.

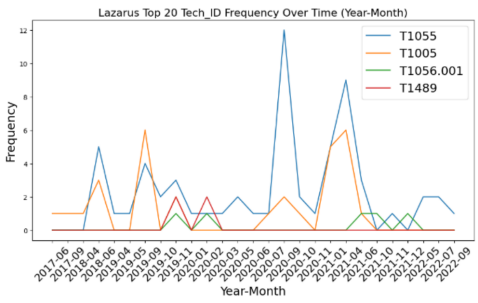

Fig. 4 Frequency of monthly use of techniques used by Lazarus among the top 20 threat score techniques.

i) Analysis of 2024 Attack Samples

In 2024, Lazarus primarily used the DLL-Side Loading technique (T1574.002) as their main attack method. This technique places a legitimate application and a malicious DLL in the same file path, causing the malicious DLL to operate simultaneously. Specifically, this was mainly used in the execution phase of the malware and the infiltration stage.

ii) Comparison with Past Statistics

It was noted that the DLL-Side Loading technique (T1574.002) showed gradual use in 2020, with a total of 12 samples identified in the data. Although this technique is not among Fig.4 used by Lazarus from 2018 to 2022, the overall statistical trend suggests it might become a favored attack method for Lazarus in 2024. This analysis can help predict future techniques Lazarus might employ and provide early insights into shifts in the group’s primary attack methods.

Therefore, by analyzing the association between past statistics and attack samples, it was confirmed that Lazarus maintains consistency in their attack techniques over time while gradually introducing new techniques.

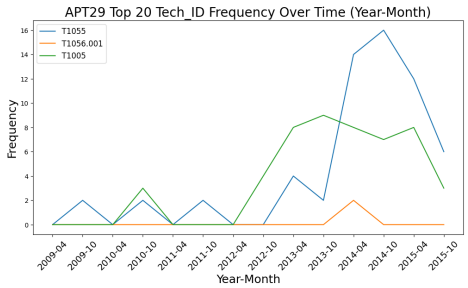

Fig. 5 Frequency of monthly use of techniques used by APT29 among the top 20 threat score techniques.

i) Analysis of 2023 Attack Samples

APT29 has been using the DLL-Side Loading technique (T1574.002) in the initial penetration stage in 2023. They design the malicious DLL to operate synchronously with the legitimate application by storing them in the same file path. In a case where APT29 spread malware via a fake vehicle ad targeting diplomats, the bmw.iso file used technique T1547.011. This involves exploiting Plist Modification and Property List Creation within an iOS environment.

ii) Comparison with Past Statistics

In the analysis based on past statistical data (2008 to 2015), T1574.002 was found 15 times. This suggests that APT29 is increasingly attempting to exploit cloud and legitimate softwares. Yet, T1574.002 did not make it to the list of top 20 threat score techniques. This indicates that the group still tends to lean on traditional techniques such as T1082 for direct system information checks, hence its absence in Fig.4.

The persistent use of techniques like T1082 is clear. By co-analyzing other statistical trends, analysts can better comprehend and anticipate shifts in attack patterns.

3.1.3 Activity Period & Attack Influence

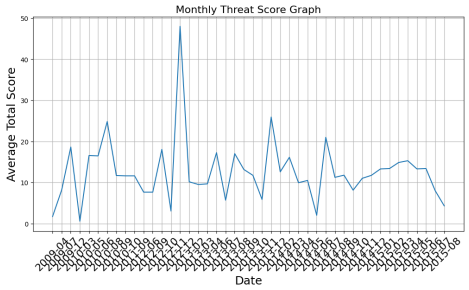

Fig. 6 Change in threat score of APT29.

Fig.6 illustrates APT29's score variation over time. Statistics analyzed, drawing meaningful conclusions proved challenging. APT29's scores remained relatively unchanged after 2013, due to the analysis of randomly limited hash values collected. In contrast, meaningful insights were found about Lazarus, with the provided access to security analysis articles. Organizations with thoroughly collected hash values may overcome the limitations such analysis. Therefore, the analysis in this section will focus solely on **Lazarus APT**.

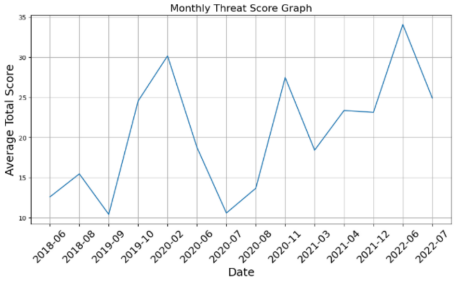

Fig. 7 Change in threat score of Lazarus.

i) Threat Score Peak in 2020 and Actual Activities

According to Fig.7, Lazarus showed the first peak of its activities from the end of 2019 to 2020. During this period, the group was known to carry out the 'AppleJeus Operation'.

ii) Analysis of Activities during the Peak Period

The Fig.7 results suggest the threat score peak in the early 2020s is intertwined with the AppleJeus Operation. In other words, the statistical peak has a direct correlation with the intensity of the attack activities. Furthermore, from January 2020, the scope of activities expanded even more. Ultimately, the trend of threat score changes indicates the probability of estimating the status of the attack behavior of Lazarus. This provides significant assistance in predicting future activities and establishing response strategies.

3.2 Insights from APT29 and Lazarus

The analysis of Lazarus revealed limitations in the proposed framework. However, it also clarified insights into the group's preferred attack techniques and potential shifts. This shows the initial analysis process of APT profiling with an automated framework. It helps maximize efficiency and refine the stages involved.

The comparison between APT29 and Lazarus allows to analyze current trends in attack techniques among APT groups. For example, statistics from this two groups show an increased use of the T1574.002 technique. This trend suggests a similar adoption likelihood among other groups.

4. Discussion

This paper analyzed Lazarus and APT29, but encountered a limitation: the omission of phishing techniques. This gap is due to the specific software environments required by document-type malicious codes, which obscure techniques like spear phishing. Despite this, an automated framework for the extraction and statistical analysis of TTPs was developed. This framework boosts analysts' efficiency and offers security professionals valuable insights. However, it also highlights areas needing improvement, especially in detecting phishing techniques and software-specific attacks. Addressing these gaps is crucial for more accurate attack pattern analyses and effective response strategies.

5. Conclusion

The proposed framework for APT profiling demonstrated its efficiency at an initial stage by analyzing two APT groups. Furthermore, by creating a system capable of automation from data collection to statistical extraction, this study has significantly enhanced the efficiency of APT profiling. In addition, a risk score was calculated for each hash and used to allocate human resources and predict attacks by APT groups.

Acknowledgement

This research was supported by the research grant from Seoul Women's University (2022-0103).

References

[1] M. Onishi, N. Hosoda, K. Nakanishi, and H. Ibayashi, "Cyber Security Operations for the Tokyo 2020 Games," *The Journal of The Institute of Electronics, Information and Communication Engineers*, vol. 105, no. 8, supplement, pp. 272–278, Augest 2022.

[2] J. Greig, "450 million cyberattacks attempted on japan olympics infrastructure: Ntt." https://www.zdnet.com/article/nearly-450-million-cyberattacks-attempted-on-japan-olympics-infrastructure-ntt/. Online; accessed 09 March 2024.

[3] B. Al-Sada, A. Sadighian, and G. Oligeri, "Analysis and characterization of cyber threats leveraging the mitre att&ck database," *IEEE Access*, vol. 12, pp. 1217–1234, 2024.

[4] K. Kim, F. A. Alfouzan, and H. Kim, "Cyber-attack scoring model based on the offensive cybersecurity framework," *Applied Sciences*, vol. 11, no. 16, p. 7738, 2021.

[5] "Virustotal." [Online]. Available: https://www.virustotal.com/.

[6] "Lazarus.day." [Online]. Available: https://lazarus.day/.

[7] "Apt malware dataset." [Online]. Available: https://github.com/cyber-research/APTMalware/tree/master/samples/APT29.

[8] S. Cho, Y. Park, K. Lee, C. Choi, C. Shin, and K. Lee, "An apt attack scoring method using mitre attck," *Journal of the Korea Institute of Information Security and Cryptology*, vol. 32, no. 4, pp. 673–689, 2022.

[9] "Lazarus 2024 sample." [Online]. Available: https://www.virustotal.com/gui/file/4df757390adf71abdd084d3e9718c153.

[10] "Apt29 2023 sample." [Online]. vailable: https://www.virustotal.com/gui/search/31867eb002d468df6ed7267d3db66a63.