UDC 004.056.5 004.272

Identification of the malicious group's digital trace using cryptography tools

Oleh Kozlenko[1], Yuliia Nakonechna[1], Mykhailo Mokhonko[1]

[1] *National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", Kyiv, Ukraine*

Annotation

Every year, information about a new data leak or compromise of a public or private organization becomes more commonplace in everyday life. The most dangerous and effective in this field are special hacker groups whose funding is associated with special government agencies or services. The study of the activities of these groups has led to identification of each unique method (or tactics, techniques and procedures - TTP) and systematization of the findings. The advantage of creating a digital fingerprint of APT groups is to quickly identify similarities in TTPs and compare these intervention attempts with known groups or compare the means of existing groups with new ones for which there is little information.

Keywords: Cybersecurity, APT, Digital Fingerprint, TTP, Cryptography

Introduction

In recent years, we've witnessed numerous occurrences of governmental and corporate information systems falling prey to attackers driven by personal motives or affiliated with foreign intelligence services. These groups each possess their unique sets of tactics, techniques, and procedures (TTPs), typically enabling their identification. Over time, research on cybercriminal behavior has been meticulously analyzed and compiled by MITRE into a knowledge base known as MITRE ATT&CK. Currently, this repository comprises 14 tactics, 188 techniques, and 379 sub techniques available for employment by cybercriminals or cybersecurity researchers. Furthermore, the MITRE ATT&CK database encompasses information on 129 groups whose activities have inflicted harm upon corporations, government agencies, or private organizations. While many TTPs for attack and post-exploitation are shared among known groups like FancyBear, DarkHalo, and OceanLotus, each group may employ its own arsenal of malware applications, communication control tools, and dedicated Command-and-Control (C2) servers. However, there's a significant likelihood that multiple software tools are shared across various groups, forming a distinct "fingerprint." The concept of entity fingerprinting in the information landscape isn't novel. Organizations utilize various methods to fingerprint a user's device, such as their browser, for commercial identification purposes. Moreover, there are instances of fingerprints being utilized in cybersecurity, like JARM, an application designed to ascertain TLS connection fingerprints. One notable advantage of employing JARM is its assistance in identifying malicious C2 server connections and creating a fingerprint for subsequent analysis and application, such as in firewalls or network filters. Currently, there exists no analogous tool or application for fingerprinting to identify malicious hacker groups, rendering the issue pertinent. In this research endeavor, the primary objective is to devise a fingerprinting mechanism for presently active groups, allowing for rapid comparison of fingerprint disparities without reliance on software applications [1].

General information about malicious groups

The Cybersecurity and Infrastructure Security Agency (CISA), a government agency, diligently monitors cyber incidents transpiring within the United States. These reports frequently entail details concerning incidents instigated by cybercriminals affiliated with intelligence services from various nations. Presently, the CISA website categorizes these incidents into four countries: China, Russia, North Korea, and Iran. Delving deeper into each country's section reveals specific incidents and associated Advanced Persistent Threats (APTs).

APTs, an acronym for Advanced Persistent Threats, represent the most perilous cybercriminal groups. These entities are distinguished not only by their intricate methods of assaulting information systems but also by their strategic objectives. Typically, these groups target infrastructure facilities, sometimes critical ones, the compromise of which could yield severe repercussions. Although cybersecurity is typically a crucial component of infrastructure facility operations, attackers continually devise novel methods and techniques to infiltrate information systems. Each cyber incident is characterized by a distinct set of techniques and software tools employed to facilitate the breach [2].

While numerous techniques and applications are recurrent across various cybercrime cases, some are unique and specific to certain cybercriminal groups. For instance, the Cobalt Strike software is frequently utilized in scenarios necessitating an agent structure for issuing commands from the primary Command-and-Control (C2) server to a Windows-based target device. Conversely, a series of AppleJeus bootloaders, employed in applications related to cryptocurrencies, has been exclusively utilized by the Lazarus Group, associated with North Korea's intelligence services and classified as an APT. Different cybersecurity organizations may assign various names to malware variants, as they often observe these activities concurrently and label the software according to organizational policies. Given the rapid pace of development in newer and more sophisticated malware, definitively identifying a malicious group solely based on the usage statistics of a particular set of malware is implausible.

It's worth noting that the quantities of techniques and tactics are subject to change and are not constant. The functionalities of programs, operating systems, and specialized device software continually evolve, with new features emerging and existing functionalities undergoing alterations. These changes introduce additional threats to system security, broaden the attack surface, and furnish attackers with more information about the system for exploitation. Consequently, new techniques, applications, and malware are developed to capitalize on the latest threats resulting from these updates. Furthermore, the disclosure of 0-day vulnerabilities may give rise to additional techniques. 0-day vulnerabilities, along with 0-day exploits, represent a unique category of vulnerabilities and exploits that leverage vulnerabilities that may or may not be known to software developers, and for which no patch exists to mitigate the vulnerability. Importantly, it's crucial to acknowledge that 0-day vulnerabilities cannot engender new tactics.

MITRE has meticulously cataloged the tactics, techniques, and software employed by various hacker groups. It's worth mentioning that different groups may share similar activity patterns or nearly identical operations. In such instances, discerning whether they belong to the same group or are simply emulating each other's activities poses a challenge. One of the objectives of this study is to devise an algorithm capable of swiftly determining, without reliance on software, whether the operational methodologies of two or more groups exhibit similarities. This capability would facilitate the prompt identification of differences in methods or ascertain whether disparities exist among several criminal hacker groups.

APT fingerprint algorithm challenge

The selection of APT groups for analysis was primarily motivated by their involvement in some of the most significant recent cyberattacks and breaches. The compromise of SolarWinds' systems, notably, was deemed one of the most impactful attacks on autonomous systems up to 2020. Brad Smith, President of Microsoft in the United States, described this attack as "the largest ever" in an article for Politico. Microsoft assigned the nicknames SUNBURST and SUPERNOVA to the malware samples discovered after analyzing the compromised systems. FireEye's analysis revealed that the attackers implanted malicious code into the ORION update program, enabling remote access to victims' systems. This exploit capitalized on a backdoor in the SolarWinds library. The legitimacy of the program certificate masked the presence of the malicious code post-update. Additionally, it is believed that a vulnerability in SolarWinds' FTP server, which had a simple access password of "solarwinds123," facilitated the download and distribution of files among the company's software users [3].

Reports and code analysis of the SUNBURST ransomware, also known as Solorigate, have implicated the APT29 group in this attack. This group's involvement extends beyond this incident, with APT29 and APT28 being linked to various other attacks, including the 2016 breach of the US Democratic National Committee servers, phishing campaigns against non-governmental organizations, attacks on government agencies in Norway and the Netherlands, and the distribution of malware such as PolyglotDuke, RegDuke, and FatDuke. Furthermore, they were responsible for the recent attack on the US Republican Committee in July 2021, stemming from the well-known compromise of Kaseya software.

In broad terms, a mapping algorithm transforms various types of data (e.g., files, connection information, objects) into a significantly condensed sequence of bytes, thereby uniquely representing the original dataset. However, it's important to note that the process of generating a data fingerprint is irreversible; the original dataset cannot be reconstructed from the fingerprint. Data fingerprints find application in scenarios such as filesystem searches, network data verification, data deduplication, and unwanted data filtering, among others.

Data fingerprinting algorithms, besides considerations like conversion speed, memory utilization, and code complexity, must ensure a unique outcome. This means that no two pairs of data subjected to the fingerprinting algorithm should yield identical fingerprints. However, when the initial source, which typically has a larger default size than the result produced by the fingerprint algorithm, is represented, collisions may occur. A collision arises when two distinct datasets produce the same result. Present-day algorithms, though, are designed to be collision-resistant, meaning the likelihood of collisions occurring is minimal.

Moreover, data fingerprinting algorithms should exhibit an avalanche effect, wherein altering a single bit of data triggers a comprehensive change in the algorithm's result. The following functions can be used to determine the data fingerprint:

- Hash functions, also called fingerprint or digest functions, are the main ones for fingerprinting. One of the important characteristics of hash functions is cryptographic strength. The most popular algorithms are MD-5, SHA-1, SHA-256, SHA-512 and others [4].
- The Rabin fingerprint scheme is an algorithm for determining fingerprints that uses polynomials over a finite field [6]. The basic idea of this method is to represent an n-bit message as a polynomial of degree n-1 over a GF(2) field:

$$f(x) = m_0 + m_1 \cdot x + \ldots + m_{n-1} \cdot x^{(n-1)}$$

Then, an irreducible polynomial p(x) of degree k over GF(2) is chosen, which defines the message fingerprint m as the remainder of dividing r(x) of polynomials f(x)/p(x) over the finite field GF(2) and can be represented as a polynomial of degree k-1 or as a k-bit number.

- Machine learning-based algorithms require a lot of data to train the system and can be used to predict future data [5].
- Locality-sensitive hashing (LSH) is a method using probabilities to reduce the dimensionality of data. These functions are characterized by a metric space M = (M,d), a threshold R>0, an approximation factor c>1, and probabilities P_i, and must satisfy the following conditions:

$$if\ d(p,q) \leq R, then\ h(p) = h(q)p\ with\ \mathrm{P}_1\ probability$$

$$if\ d(p,q) \geq cR, then\ h(p) = h(q)\ with\ P_2\ probability$$

The most popular implementations of this family of algorithms are MinHash and SimHash, which are used to determine the similarity of two data sets [5].

- Combined variant - a combination of two or more data fingerprinting algorithms to obtain an aggregated fingerprint.

All parameters pertaining to group activities were sourced from the MITRE ATT&CK framework, designed to address the identification of system and device compromise methods.

One of the interesting examples of the use of fingerprints in the field of network cybersecurity is JARM [7], which is a continuation of the previous similar application JA3. The main idea of this application is to send 10 TLS Client Hello packets to determine a unique set of responses, which is then aggregated and hashed using a special algorithm that produces the final fingerprint of the connection. The final fingerprint consists of a 62-character fingerprint, which in turn is a composite of a 30-character block containing data such as the TLS version, selected ciphers, and a 32-character block of the truncated SHA256 hash result of the main extensions on the server. In this way, individual servers can be identified by their already compiled fingerprints, and it is possible to determine whether a given server belongs to a family of other devices with a similar fingerprint.

Cybersecurity professionals can use JARM to detect potentially malicious activity that threatens the environment. This is especially true for popular attack frameworks such as Cobalt Strike and similar C2 infrastructure that are difficult to defend against.

APT fingerprint method proposition

Let's analyze the options for the fingerprint function to determine the most suitable one for the task at hand.

- The use of cryptographic hash functions such as MD-5 and the SHA family. One of the tasks is the ability to quickly determine the similarity of activities of malicious groups among themselves without the use of software applications. The avalanche effect inherent in these hash functions makes it impossible to determine the similarity, but only to monitor whether changes have occurred. These functions are not suitable for a general dataset, but can be used as part of a combined approach.
- Using a Rabin diagram for a dataset. A feature of the Rabin scheme is its effective use in text data types, as well as its resistance to collisions. This algorithm also has an avalanche effect, which does not allow you to quickly determine the similarity of activities or aspects of the group's activities. It can be used in a combined approach.
- The use of machine learning is a very effective method, but currently there is not enough information to train the network, which makes the method difficult to implement.
- Locality-sensitive hashing (LSH) allows comparing data with each other and determining their similarity. This property is extremely important for the task at hand, and therefore the use of this method is a priority for the overall data set, but not for analyzing parts of the activity.
- Combined approach. As you can see, different methods have advantages in some aspects and disadvantages in others. Using several variants of fingerprint functions for different aspects of fingerprinting will allow you to fulfill the conditions of the task, such as comparing aspects of group activities and presenting information about groups in a unique, abbreviated form.

All the parameters of the groups' activities were obtained from the MITRE ATT&CK framework, which was created to solve the problem of identifying ways to compromise systems and devices. To solve this problem, we need to find a way to represent the activities of malicious groups in the form of, for example, a hash string that will be unique to each group and will uniquely identify the corresponding group.

To begin with, let's define how the data will be presented. Let's take for example how the MITRE ATT&CK framework presents information about APT-16, which is stored in JSON format:

{ "description": "Enterprise techniques used by APT16, ATT&CK group G0023 v1.1", "name": "APT16 (G0023)", "domain": "enterprise-attack", "versions": { "layer": "4.2", "attack": "10", "navigator": "4.3"}, "techniques": [], "techniqueID": "T1584", "showSubtechniques": true}, {"score": 1, "techniqueID": "T1584.004", "showSubtechniques": true, "comment": "[APT16](https://attack.mitre.org/groups/G0023) has compromised otherwise legitimate sites as staging servers for second-stage payloads.(Citation: FireEye EPS Awakens Part 2)"], "gradient": { "colors": ["#ffffff", "#66b1ff"], "minValue": 0, "maxValue": 1}, "legendItems": [{"label": "used by APT16", "color": "#66b1ff"}]}

As you can see, the information concerns only the techniques used by this group, without information about the available malware applications, although they are contained in the relevant section on the web resource (in this case, only S0064, ELMER). As we noted earlier, a significant characteristic of the attackers' activities is the tools they use, not just a set of tactics and techniques. Also, any information that does not relate to tactics, techniques and tools is unnecessary for this task. In this case, the information in JSON format that will meet the requirements of the task will look like this:

{ "name": "APT16", "domain": "enterprise-attack", "techniques": [{"techniqueID": "T1584.004", "title": "Compromise Infrastructure: Server"}], "softwares":[{"softwareID": "S0064", "title": "ELMER"}]}

This representation takes into account the main aspects of the attackers' activities according to the information about them in the MITRE ATT&CK framework and can record additional information if necessary (for example, the IP address of C2 servers used by attackers in Cobalt Strike configurations).

A special feature of the representation is separation into the main parts of the array (name, domain, softwares, techniques) and the ability to update with additional information. As an example, you can add arrays of IP addresses of C2 servers used by attackers.

The above analysis of fingerprinting functions allowed us to form an idea of what functions will be used to create a fingerprint of APT groups. In this study, the authors of the article decided to use a combined approach to creating a function. This step is due to the fact that information about the group's activities and the area of its operation is presented in the form of independent sets and, accordingly, one part of this information can be hashed by a different function than the other.

The following algorithms were chosen to solve the task:

1. MD5 - for hashing the domain section;
2. SHA-224 - for hashing the software section;
3. SHA-256 - for hashing the techniques section;

The "name" field will not be hashed and will be used as a fingerprint identifier.

For the example of obtaining a fingerprint, let's take the information about APT-16, as presented earlier. Let's break it down into parts:

- "name": "APT16"" - will be used as an identifier
- "domain": "enterprise-attack"
- "techniques": [{"techniqueID": "T1584.004", "title": "Compromise Infrastructure: Server"}]
- "softwares":[{"softwareID": "S0064", "title": "ELMER"}]

So the resulting APT-16 group fingerprint function will look like this:
3468f22d494c679d74f38e463221fb83fa8d0f27353f543a94d241e667626510bc69be316c3d223d614acd1ca8f3b5ff27f226b666c5e69e8375164f06931753b1ca343210e22227906e27f2

In parts, this print will be presented as:

- enterprise-attack - 3468f22d494c679d74f38e463221fb83
- [{"techniqueID": "T1584.004", "title": "Compromise Infrastructure: Server"}] - fa8d0f27353f543a94d241e667626510bc69be316c3d223d614acd1c
- [{"softwareID": "S0064", "title": "ELMER"}] - a8f3b5ff27f226b666c5e69e8375164f06931753b1ca343210e22227906e27f2

This approach will help to achieve one of the goals of the project - the ability to monitor changes and determine how similar the prints of different groups are without using additional software.

To implement the functionality of determining a more detailed assessment of the similarity of two prints, we used the SimHash methodology, which was proposed by Moses Charikar to determine the similarity of two data sets.

In the process of working on the study, we also set a goal to develop a utility to perform the assigned tasks. All of the above methods and functionality were implemented and programmed in Rust. The resulting utility is available on the GitHub repository at https://github.com/lxldx/A2PTF.

Conclusion

In this study, the authors analyzed existing algorithms for creating digital fingerprint functions, analyzed APT groups, and information from open sources, primarily from the MITRE ATT&CK framework, which can be used to determine the fingerprint of group activities. This work resulted in the proposed fingerprinting function, which consists of 3 main parts: MD5 for hashing the domain section, SHA-224 for hashing the software section, and SHA-256 for hashing the techniques section. This division can help to quickly determine what both groups have in common. Also, for a more specific comparison, it was proposed to use the SimHash function to determine the l0evel of similarity between two fingerprints. The result of the effort is a software application that allows generating a group activity fingerprint from a JSON file.

References

[1] "What Is an Advanced Persistent Threat (APT)?". Cisco. Retrieved 11 August 2019.

[2] MITRE ATT&CK framework - https://attack.mitre.org/

[3] Gonzalez, Joaquin Jay, III; Kemp, Roger L. (16 January 2019). Cybersecurity: Current Writings on Threats and Protection. McFarland. p. 69. ISBN 9781476674407.

[4] Kleppmann, Martin (2 April 2017). Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems (1 ed.). O'Reilly Media. p. 203. ISBN 978-1449373320.

[5] "Proposed Revision of Federal Information Processing Standard (FIPS) 180, Secure Hash Standard". Federal Register. 59 (131): 35317–35318. 1994-07-11. Retrieved 2007-04-26

[6] Broder, Andrei. (1998). Some applications of Rabin's fingerprinting method. 10.1007/978-1-4613-9323-8_11.

[7]. JARM - https://github.com/salesforce/jarm