On the assessment of completeness and timeliness of actionable cyber threat intelligence artefacts

Cagatay Yucel, Ioannis Chalkias, Dimitrios Mallis, Evangelos Karagiannis, Deniz Cetinkaya and Vasilios Katos

Bournemouth University, Fern Barrow, Wallisdown, Poole BH12 5BB
{cyucel,ichalkias,dmallis,ekaragiannis,dcetinkaya,vkatos}@bournemouth.ac.uk

Abstract. In this paper we propose an approach for hunting adversarial tactics technics and procedures (TTPs) by leveraging information described in structured cyber threat intelligence (CTI) models. We focused on the properties of timeliness and completeness of CTI indicators to drive the discovery of TTPs placed highly on the so-called Pyramid of Pain (PoP).
We used the unit42 playbooks dataset to evaluate the proposed approach and illustrate the limitations and opportunities of a systematic intelligence sharing process for high pain TTP discovery.

Keywords: ATT&CK framework, cyber threat intelligence, unified cyber kill chain, pyramid of pain, data quality dimensions.

1 Introduction

The continuous evolution and adaptability of cyber threat actors reflected in their fluid modus operandi is mandating radical changes in the cybersecurity sector. At the same time, the constantly increasing number of cyber-physical and IoT devices, the use of new tools with extended capabilities and devices with limited consideration to security bring forth new attack vectors that are reshaping the threat landscape [1], introducing new or advanced threat actors. Advanced Persistent Threat (APT) groups with enhanced means and recourses to cause significant damage to operations both in private and public sector cannot be confronted by the standard incident management mechanisms [2] and require new approaches in organizing cybersecurity in the technical, tactical, operational and strategic level [3].

Information sharing offers the potential of building a trusted network of partners with the purpose of circulating cyber threat related information in order to raise awareness for newly discovered cyber threats and provide with solutions to already known security issues. This action of information sharing is encumbered with mitigating factors that constitute the process challenging. It requires a coordinated approach from partners that could enable a common culture in terms of the sharing practice. The amount of data produced and consumed by organizations is constantly increasing, forcing them to adopt automation in order to analyze and evaluate threat information over a number of properties such as its value, relativity, context, timeliness and ingestibility

[4]. By overcoming those challenges, the organizations that participate in the information sharing constituencies manage to gain information about threats and attacks on their information systems, possible countermeasures, patches on vulnerabilities and other types of incident response and systems protection that could also increase their readiness levels.

In this research, we contextualized our approach around PoP in order to evaluate quality in threat intelligence sharing [5]. A CTI that contains information about a hash or an IP, might be mapped low in the PoP but it will serve critical purpose in order to address an urgent incident in a time that can be considered acceptable in terms of incident response. As long as the levels of completeness of the CTI increase (i.e. the CTI contains information about the payload, the attack patterns and the actor) the CTI is mapped higher to the PoP and the information shared offers a more complete description of the attack which has higher value.

The mapping to the PoP allows us to gain a better understanding regarding the level of the CTI (technical, tactical, operational, and strategic). CTIs with information that hangs low in the PoP belong to the technical level, where information is provided with the forms of Indicators of Compromise (IoCs), Indicators of Attack (IoAs), forensic evidence and technical description. The tactical level contains information related to the upper levels of the PoP which are harder to obtain; tools and TTPs offer context to the analysis of the attack and provide information regarding the actors, thus increasing the completeness of the report, with time being the trade-off of that case. For the operational and strategic levels of threat intelligence a higher-level analysis of data is required. The information derived from such analysis has the potential to offer not only incident managing solutions for post-mortem or live isolated artefacts but also to predict impending attacks or analyze attack behaviors in enterprise level.

The relationship between PoP and the levels of CTI can be defined by timeliness and completeness. The two dimensions have an inversely proportional relationship that can be illustrated with the use of a kill chain. The discovery of an incident at the early stages of a kill chain provides with a timely incident detection; the indicators lie in the low levels of the PoP and threat intelligence levels addressed are the technical and operational level. On the other hand, when an incident is detected at one of the later stages of a kill chain, the event is mapped on the higher level of the pyramid of pain, since its detection is more onerous. The dimension of completeness is reflected by the number of mapped stages in the kill chain.

The increased number of the identified stages in the kill chain provides with adequate information of the organizational and strategic levels of threat intelligence and possibly, an understanding of the timeline of the investigated attack. For this reason, we propose a novel approach of constructing a timeline using low pain indicators in order to reach to TTP level of the PoP.

The paper is organized as follows. Section 2 discusses the landscape of challenges that complicate the process of sharing threat intelligence. Section 3 introduces the proposed approach and methodology. Evaluation is discussed in Section 4. Finally, conclusions are drawn, and future work is suggested in Section 5.

2 Challenges of Cyber Threat Intelligence

Cyber threat intelligence (CTI) sharing has been highly acknowledged in the last decade as a promising answer to the ever-increasing complexity of cyberattacks [1, 6, 7]. Despite the possible benefits, CTI producers and consumers are facing several issues and challenges [8].

At the top of this list is the Threat Data Overload; the first and most common struggle for those who try to acclaim the benefits of the CTI sharing process and are actively involved in it. The aim of the Threat Intelligence Sharing Platforms (TISPs) is to manage CTI data and feed consumers and threat management teams with actionable information. However, there are still lots of limitations and obstacles to achieve that goal. A recent survey among 1200 IT and IT security practitioners, showed that most respondents are partially or not satisfied at all regarding the CTI feeds they receive [9]. For example, high percentages (from 30% up to 70%) reflect the significant inadequacy of CTI information in terms of relevance, timeliness, accuracy, completeness, and ingestibility; which, according to the European Union Agency for Network and Information Security (ENISA) , are the five criteria an information should meet to be actionable and support decisionmakers [10].

The Data Quality (DQ) of the shared feeds is also among those challenges, with implications in the decision-making processes which mainly derive from the fact that data quality is inherently subjective and directly related to the actionability of information [10]. These criteria are used to measure the quality of data and further evaluate the available threat intelligent feeds (i.e. mainly public blacklists) in the literature [11]. The same approach was followed by M. Faiella et al. [12] to define the "weighting criteria" and use them as part of the proposed Threat Score (TS) function; a function used to evaluate IoCs collected from various sources in order to support Security Operations Center (SOC) analysts to prioritize the incidents' analysis. However, this was only a part of their overall contribution towards enriching threat intelligent platforms capabilities. Also, the study in [4] investigates the qualities (i.e. timeliness, completeness, robustness) of IoCs collected by several open sources in order to understand how effective these sources are.

The challenges of data quality in threat intelligence sharing platforms were also investigated by Sillaber et al. [13]. Aiming to address the factors affecting data quality of CTI at each of the following levels (i.e. gathering, storing, processing, and sharing data), the authors conducted studies, starting from interviewing several security-related stakeholders operating within international organizations. Their analysis was based on five traditional data quality dimensions which include accuracy, completeness, consistency, timeliness, and relevance respectively. As a result of that research they presented various findings and recommendations regarding threat intelligence data quality. The authors of [14], focused on the topic of quality of data generated by incident response teams during investigations. Their methodology was based on a case study within a financial organization to empirically evaluate data quality. During the second phase of data gathering they conducted analysis in terms of the accuracy, timeliness, completeness, and consistency of the collected data. According to this analysis, there is still a lot of future work to be done towards enhancing the quality of data generated by

incident response teams in order to facilitate and support CTI. The same metrics were suggested by S. Sadiq in his handbook [15] under the category of data values which is one of the three main categories defining the dimensions of data quality.

The list of challenges also consists of issues related to Privacy and Trust, as information disclosure and the potentially negative after-effects are big concerns for the companies involved in CTI sharing. In addition, diverse data models, tools and standards are used to exchange threat intelligence feeds impose Interoperability issues to TISPs. For that reason, the aim is to standardize the way threat intelligence platform vendors and open-source CTI developers communicate cyber threat intelligence data. Among the foreseeable benefits, is the improvement of analytical and management capabilities which will further increase the value of the shared CTI.

3 Proposed Approach

3.1 Justification of the Methodology

As mentioned, CTI data can be incomplete, unreliable or subjective. At the same time, a significant amount of work has been invested into standardizing and harmonizing the description and sharing of CTI data. In this section, we propose an approach to assess the quality aspects of CTI data via mining and exploring the captured information, and to evaluate the quality of the underlying information sharing scheme while identifying areas for improvement. As such, in order to assess the quality aspects of CTI data, we propose an approach that builds upon a widely accepted threat intelligence sharing standard and show how though mining and exploring the captured information we can evaluate the quality of the underlying information sharing scheme as well as identify areas for improving it.

Our proposed approach builds upon a widely accepted threat intelligence sharing standard, more specifically, we sample a number of incidents and campaigns that have been studied and expressed in a STIX notation. Although a STIX diagram may effectively reveal relationships between the different objects (such as threat actors, victims, campaigns, IoCs and so forth), it does not show the sequence of the underlying attack vector, although timeline information may be included as a field in the object properties. We conjecture that timing information is critical when extracting TTP descriptions, as this is inferred and assumed in other threat modelling approaches such as the cyber kill chain. For instance, since APT type of attacks involve more than one stages and are prescribed through a sequence of actions, we explore how timeline information can trigger the understanding and extraction of the TTP descriptors. Moreover, this information will be used not only to show the sequence of an attack, but also to use attack tree views to express the TTP. This will also allow the identification of bottlenecks in the attack sequence and the establishment of highly disruptive security controls and remedies, yielding the highest "pain" to the threat actor.

The overall approach is described in Figure 1. Starting with a given case study, it is assumed that a group of expert security analysts have extensively studied the case and developed a narrative describing the incidents in the most accurate manner. As such,

the narrative constitutes the ground truth. Following the free text description, a number of modus operandi are extracted and enriched with IoCs in order to construct the STIX model.

In this work, we use the STIX model and respected narrative as input to the proposed process. More specifically, we first consider the STIX model and use this to generate two models, a timeline of events and an attack tree. The timeline is constructed from the available timestamp information contained in the STIX objects. In addition, we leverage the standardized description of the threat actors' tactics by referencing the incidents using the ATT&CK framework. This also helps to map the events to the Cyber Kill Chain. In order to be as inclusive as possible, we adopt the unified kill chain [16] that can in principle cover for any foreseeable permutation of the dataset.

The kill chain also helps to evaluate the quality of the STIX model as well as the correctness of the examined data captured in the STIX object properties. As the kill chain suggests a loosely coupled – yet in some cases strict – sequence of events and attack patterns, we can evaluate the correctness of the captured data. Formally, given an alphabet S representing elements of an attack set (that is, standardized attack techniques or tactics), we define the range of all possible *valid* words that compose the attack space A_S , that is, all actions that can be potentially observed as an attack vector. The attack space is a subset of the free monoid S^*, under the word concatenation operation. By following such convention, A_S would be constructed following a number of rules using formal expressions that leverage the field of combinatorics on words. In essence, the informal description of the rules will still be provided by the expertise and knowledge found in the cybersecurity domain, but the modelling will be formally constructed leveraging the aforementioned field of theoretical computer science.

An example of such an arrangement is as follows. Let S contain elements of a cyber kill chain phases. The number of phases would be $|S|$. Let u be a word describing a sequence of phases that need to proceed an attack pattern described by the phases of word v. Then, a valid attack w containing v, is the one where u is the prefix of v, that is $u = vw^{-1}$.

Let w_R be the actual attack and w_T the attack pattern specified by the generated timeline. w_R is a member of A_S, but for w_T the following can hold:

- $w_R = w_T$. In this case the timeline describes the attack accurately, and the quality of the CTI can be considered as high;
- $w_R \neq w_T$ and $w_T \in A_S$. In this case the timeline describes a valid attack, but not the actual one. The CTI can be considered to be of a medium quality;
- $w_R \neq w_T$, and $w_T \notin A_S$. In this case the timeline describes an invalid, unrealistic type of attack and the CTI can be considered to be of a low quality.

Moreover, in the case where $w_R \neq w_S$, a distance metric can be applied in order to establish the proximity of the timeline CTI to the actual attack, as described by the narrative.

This research also leverages attack trees in order to enrich the description of an attack and identify potential TTPs. In a general setting, attack trees have significant drawbacks mainly due to the complexity of attacks, making them an uninviting tool for security assessment and management exercises. Attack trees can be effective when the complexity is relatively low, or when we study a particular attack subset. As such, whilst

attack trees are not suitable for security assessments, they can be employed in post-mortem analysis of security incidents, following the investigation and to contribute to the "lessons learned" phase of the incident response process [17]. By doing this, we argue that TTPs will be evident in the generated attack trees, as they will describe the actual attack rather than some instance of a likely attack.

3.2 Methodology

The followed methodology for this research starts with importing the STIX documents given in the dataset section. STIX v2 [18] documents are composed of entries from the types of indicators, observables, attack patterns, incidents, threat actors, reports, campaigns, exploit targets, packages, course of actions, TTPs (tactics, techniques and procedures). The dataset analyzed in this research contains 25 STIX files constructed from campaigns of advanced persistence threats (APTs); regarding the intelligence data, these files contain indicators formed of the malicious executables, payloads, malicious websites and IP addresses. These indicators are connected to the attack patterns from MITRE's ATT&CK framework through relationships within the CTI data.

As explained in the challenges section, the indicators given in the CTI file are considered as low-pain intelligence data. It is relatively easy for an attacker to evade the security measurements generated for these indicators comparing to the case where the TTP and modus operandi of the adversary is known. In this study, we argue that an approximation to the TTP by exploring the timeline and the relationship between indicators and attack patterns. In this study, the processed STIX documents are converted to an attack tree on the time axis and an attack vector formed from unified kill chain phases of attack patterns ordered by the timestamps of indicators. The methodology is given in Figure 1.

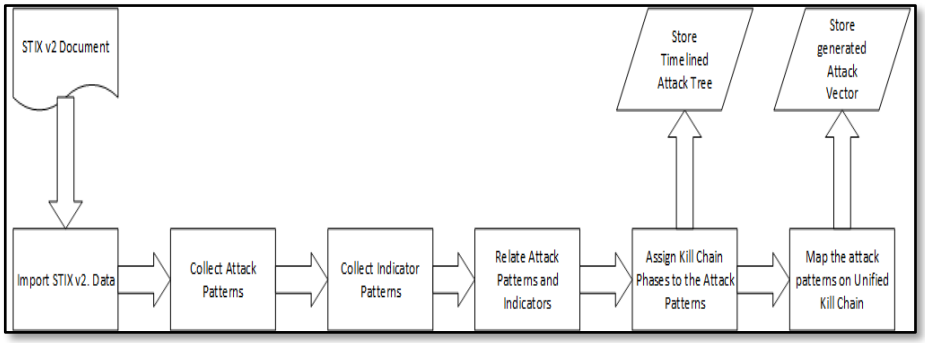

Figure 1. The General Methodology of Generation of the Timelines Attack Trees and Kill-Chain Attack Vectors

Construction of Timeline of Attack Trees and Kill Chain Attack Vectors – A showcase

For the demonstration of our methodology, this section contains the construction of the attack trees and the kill chain vectors for a simple STIX file provided within the dataset by the UNIT42 group [19]. This STIX file is a fictional one created to demonstrate their playbook viewer given in the references. The attack patterns from MITRE's ATT&CK Framework provided in this scenario is given in Table 1. This CTI document describes an attack campaign where a spearfishing (hyper)link is sent and through scripting on commonly used ports more than one communication is established to the targeted system. Indicators in this event shows that at least one screenshot is captured from the targeted system as an objective.

Table 1. Attack Patterns provided for the simple-playbook

Attack Technique Code	Description
T1192	Spearphishing Link
T1043	Commonly Used Port
T1064	Scripting
T1108	Redundant Access
T1113	Screen Capture

In the first step of the methodology, the attack patterns and indicators are filtered from the CTI document and merged. There are eight indicators given for the above five attack patterns. The indicators are as follows in Table 2. As can be seen from the table, there are three dates provided for each indicator. 'Created' date represents the timestamp when the indicator is created, 'modified' is the last modified date and the date 'valid from' represents the date that this indicator is started to be observed.

Table 2. Indicators for the simple-playbook scenario

Name	Created	Modified	Valid From
<SHA1 value placeholder>	2019-06-25 18:13:55.619	2019-06-25 18:24:15.157	2019-06-25 18:13:55.619
https://verysuspicious.com:443	2019-06-25 18:10:03.432	2019-06-25 18:24:15.157	2019-06-25 18:10:03.432
https://notsuspicious.com /givecreds	2019-06-25 18:07:17.633	2019-06-25 18:24:15.157	2019-06-25 18:07:17.633
https://dailymemes.net	2019-06-25 18:15:42.452	2019-06-25 18:24:15.157	2019-06-25 18:15:42.452
<SHA1 value placeholder >	2019-06-25 18:17:50.493	2019-06-25 18:24:15.157	2019-06-25 18:17:50.493
<SHA1 value placeholder >	2019-06-25 18:21:42.106	2019-06-25 18:24:15.157	2019-06-25 18:21:42.106
https://canhazcreds.xyz/kthanks	2019-06-25 18:23:00.359	2019-06-25 18:24:15.157	2019-06-25 18:23:00.359

As can be seen from the indicators, the attack campaign ends within approximately 20 minutes and the attack patterns containing indicators are given in the timeline. The number of indicators shows the count of the indicators grouped by the timestamps. If there are more than one type of attack pattern for a given timestamp, they are separated with a comma as can be seen from Figure 2.

In order to further map this information onto vector space, the attack patterns are vectorized using unified kill chain. The unified kill chain is a merge of kill chain standards that show some degree of recognition and adoption by the security community. As it will be explained in the next section on dataset details, the data that this proof of concept work has been implemented on is using ATT&CK and Lockheed kill chain definitions. Thus, how these two definitions are merged in this research is shown on the following Table 3. This merging implementation is adapted from the work of [16].

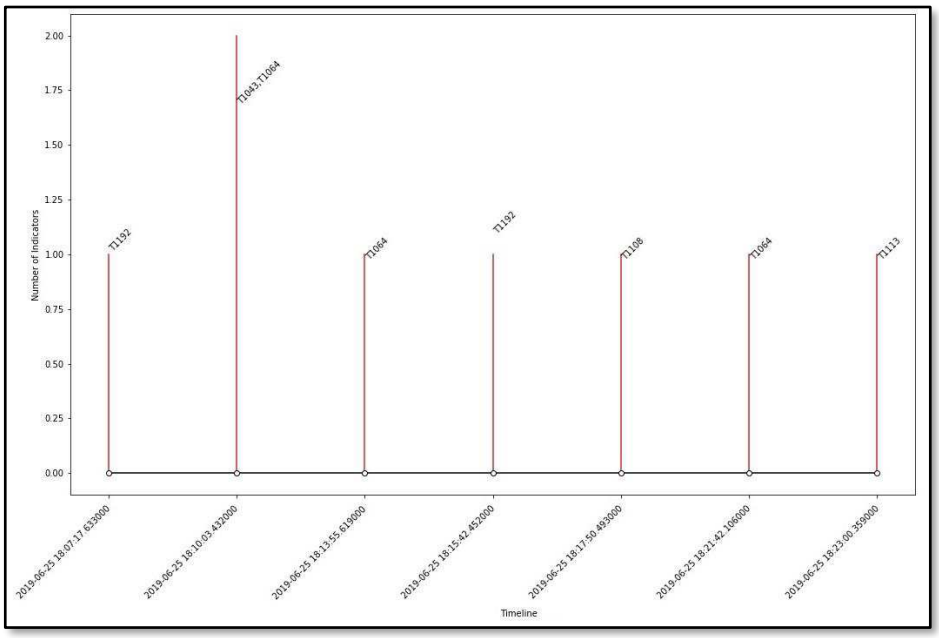

Figure 2. Resulting timeline of the simple-playbook file.

Table 3. Unified Kill Chain Mapping

ATT&CK	Lockheed-Martin	Unified Kill Chain	Vectorization
Initial Access	Reconnaissance	Reconnaissance	0
Execution	Weaponization	Weaponization	1
Persistence	Delivery	Initial Access	2
Privilege Escalation	Exploitation	Delivery	3
Defense Evasion	Installation	Exploitation	4
Credential Access	Command and Control	Execution	5
Discovery	Actions on Objectives	Privilege Escalation	6
Lateral Movement		Defense Evasion	7
Collection		Installation	8

9

Exfiltration	Persistence	9
Command and Control	Credential Access	10
Impact	Discovery	11
	Lateral Movement	12
	Collection	13
	Command and Control	14
	Exfiltration	15
	Impact	16
	Actions on Objectives	17

Hence, following this vectorization scheme, the attack vector of the simple-playbook document maps onto the following vector:

$$v_{simple\ playbook} = (2, 14, 7, 5, 7, 5, 2, 7, 9, 7, 5, 13)$$

3.3 Utilization of Levenshtein Metric

As given in the introduction, a good level of maturity is reached among the cyber threat intelligence sharing platforms on which format to share the intelligence data. However, to measure the quality of the intelligence data and analyzing its merits is still an open question. By providing an approach to mapping onto vector space and a metric integration, this study aims to extend the literature further on this ramification. For this aim, the Levenshtein Metric on vectors is chosen to be similarity metric. The pseudocode of the straightforward calculation of this metric is given as follows:

Input: $w_R, w_S \in A_S$, two attack vectors with length |r| and |s| respectively,
Output: Levenshtein distance of these two vectors

```
Procedure levenshtein(w_R, w_S):
    |r| = len(w_R) + 1, |s| = len(w_S) + 1
    matrix = Zero matrix with dimensions (|r|, |s|).
    for x in range(0, |r|):
        matrix [x, 0] = x
    for y in range(0, |s|):
        matrix [0, y] = y
    for x in range(1, |r|):
        for y in range(1, |s|):
            if w_R[x-1] == w_S[y-1]:
                matrix [x,y] = min(matrix[x-1, y] + 1,
                                   matrix[x-1, y-1],
                                   matrix[x, y-1] + 1)
            else:
                matrix [x,y] = min(matrix[x-1,y] + 1,
                                   matrix[x-1,y-1] + 1,
                                   matrix[x,y-1] + 1)
    return (matrix[|r| - 1, |s| - 1])
```

In an ideal analysis, where all the evidence and indicators are extracted in the right chronological order, we expect the order of the attack vector to have a small distance from the order of the kill chain. However, in a real-world scenario, it is rarely the case; the timelines of the indicators are not aligned in most cases. This can be witnessed not only in the CTI dataset of the Unit 42 [19] but also in other datasets as well. To converge the ideal chronological order from the kill chain phases, application of machine or deep learning algorithms over huge datasets which is out of the scope of this paper is required. Despite the limiting factors of our metric, to show our methodology, we used the order of the unified kill chain phases described in section 3b.

The timeline of the kill chain might not be aligned for several reasons. For example, in the attack patterns of muddy water CTI document, see Figure 3, the indicators begin with the identification of a communication channel between the targeted system and the malicious adversary. This is the point where the forensics investigation starts. After the point of the discovery, other indicators and evidence are collected and mapped on the timeline. In an ideal and successful incident identification and response, it is expected to trace an attack event from the beginning of the kill chain (i.e. the reconnaissance stage) or at least at the delivery stage of the kill chain and continue tracing the stages of the kill chain in the order that derives from the theoretical analysis of kill chains.

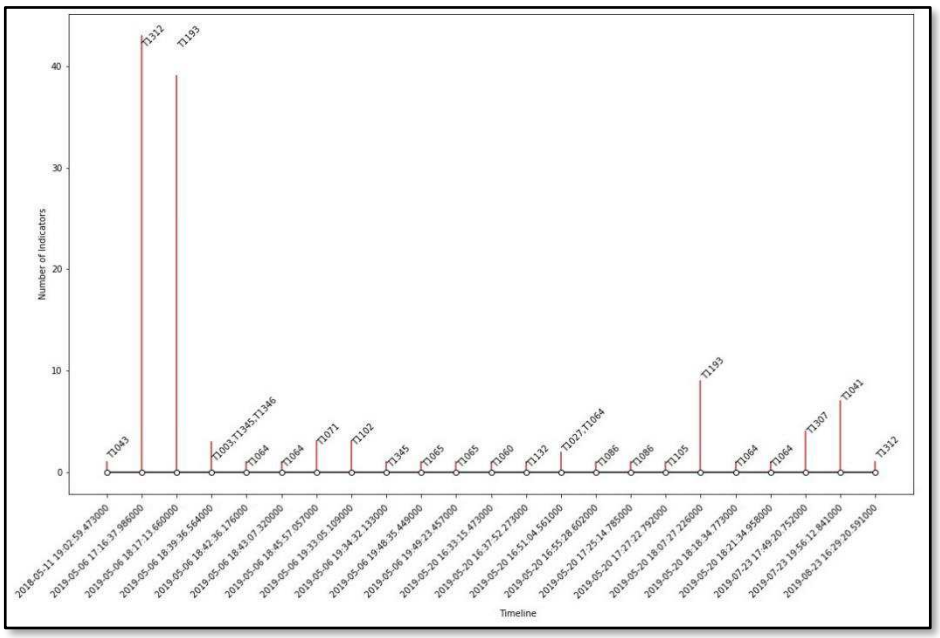

Figure 3. Timeline generated for muddy water STIX data. Note that the x labels are not placed proportionally for the sake of readability. The indicators timeline starts with the discovery of an attack pattern T1043 – Commonly used port.

The analysis utilized in this research also reflects on the data quality dimensions directly. The quality dimensions of completeness and accuracy can be considered of "higher" quality by having at least one indicator on every possible kill chain; timeliness- having started recording attackers' actions and sharing accordingly through indicators from the delivery phase; relevance and consistency – presenting only the related indicators in a CTI document and not inserting unrelated events.

4 Evaluation

4.1 Dataset

The dataset used in this research is gathered from the STIX documents created from the Palo Alto Unit 42 cyber security reports. These reports and the STIX data can be found in [19]. There are 23 CTI data provided in STIX v0.2 format. Each document is formed of several campaigns, and each campaign has a set of attack patterns described by their ids of MITRE's ATT&CK framework [20].

4.2 Findings

In this section, two examples from the dataset are inspected and interpreted. The first one is adversary group named "Scarlet Mimic" and the attack timeline is given in Figure 4.

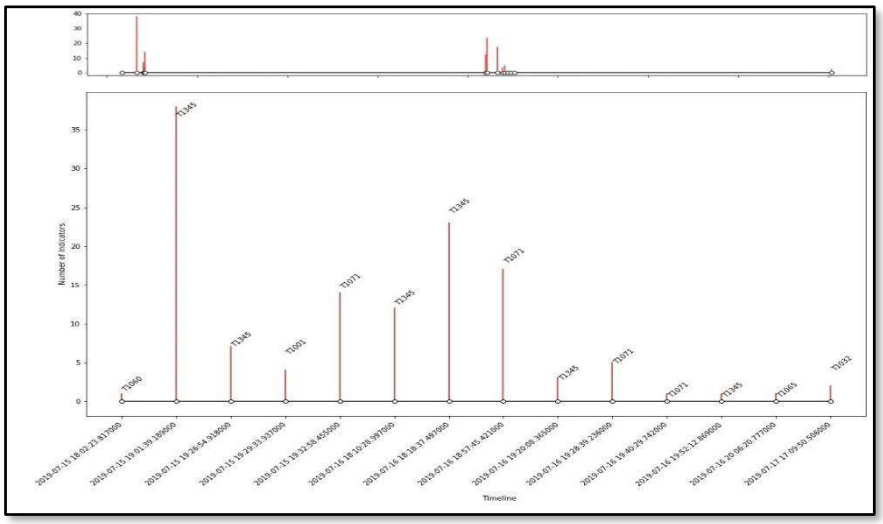

Figure 4. Scarlet Mimic Adversary - timeline. The top graph shows the frequency and distribution of the attack patterns generated with the methodology described in Section 3. The timeline is then reformatted to the graph in the lower part of the image in order to improve the readability.

By extraction of the TTP (can be seen in Table 3), we understand from the generated timeline that the analysis starts from the discovery of persistence phase of the attack pattern "T1060 – Registry Run Keys/Start-up Folder". This leads to the revealing of custom payloads which is given by the attack pattern T1345. Next step is the utilization of obfuscation in the payloads and malicious files. Their attack pattern includes usage of defense evasion techniques by leveraging the standard communication and cryptographic protocols through the uncommonly used ports.

Table 4. Extracted TTP from the timeline of "Scarlet Mimic"

T1060	Registry Run Keys / Startup Folder
T1345	Create custom payloads
T1001	Data Obfuscation
T1071	Standard Application Layer Protocol
T1065	Uncommonly Used Port
T1032	Standard Cryptographic Protocol

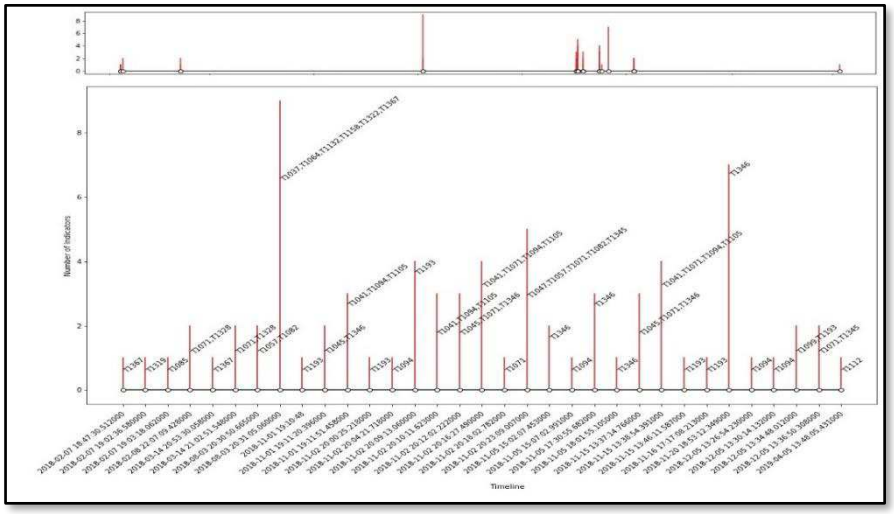

Figure 5. Sofacy's Timeline

By analyzing the chronological order of the events, we can deduce the timeline from the attack pattern. This analysis gives us an overview of Sofacy's deployment. As can be seen, the adversary initiates the attack by sending an email which contains a malicious code (T1367, T1193). Additionally, the tactic id, T1319, identifies the fact that the script in the malicious attachment is obfuscated. This is a common technique to avoid AV detection. The malicious code affects the .dll Rundll32 (tactic id T1805). The purpose of the Rundll32 is to load and run 32-bit dynamic-link libraries (DLLs). From this attack pattern code, we can understand that some piece of malicious code is being installed in the infected computers. From the attack ID T1071 we conclude that the malware uses the application layer protocol (OSI layer 7 applications, such as

HTTP, HTTPS, SMTP, or DNS) to communicate with the adversaries' command and control servers (T1094). Finally, with the information given from the attack pattern codes T1346 and T1112, we can observe that the malware code has features that render it persistent. The above information can offer to the analyst a quick overview of the Sofacy's malware features and operations in order to respond in a more organized and accurate manner, in case of an incident.

Table 5. Extracted TTP from the timeline of "Sofacy"

T1367	Spearphishing messages with malicious attachments
T1319	Obfuscate or encrypt code
T1085	Rundll32
T1193	Spearphishing Attachment
T1094	Custom Command and Control Protocol
T1071	Standard Application Layer Protocol
T1346	Obtain/re-use payloads
T1112	Modify Registry

By utilizing the knowledge of the list of techniques (Table 4 and Table 5), the timeline and any constraint rules, we can obtain the following representations of TTPs in the form of an attack tree. The root of the tree is the goal which is set as the "deepest" observed phase of the kill chain (in this case Command and Control), whereas the leaves are the observed techniques prioritized through valid kill chain sequences (or words). Figure 6 illustrates the attack trees for both playbooks, Scarlet-Mimic and Sofacy.

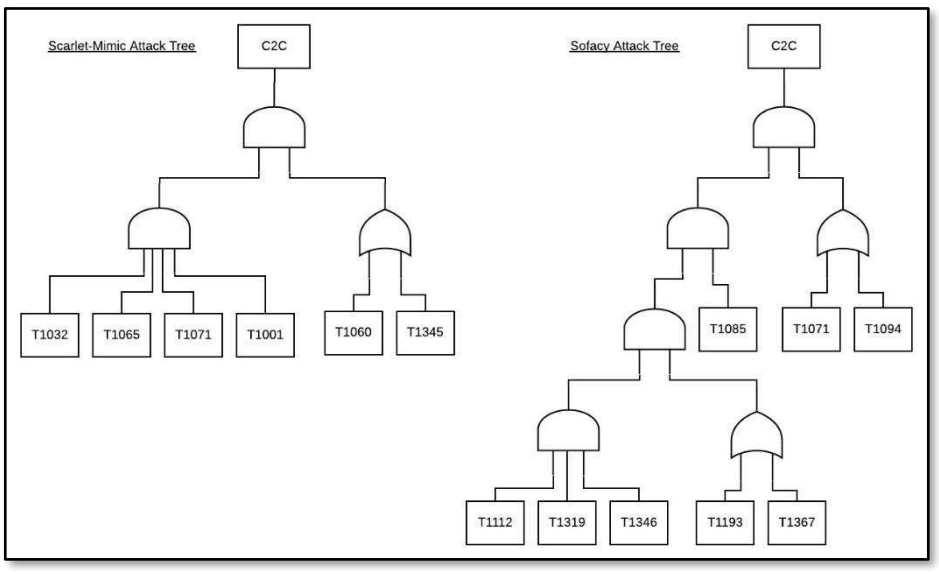

Figure 6. Attack trees for Scarlet-Mimic and Sofacy playbooks.

Finally, we apply the Levenshtein algorithm on each attack in order to calculate the distance between the attack vector constructed from the kill chain phases and the order of the unified kill chain (Figure 7). The bigger the distance of the attack vector from the unified kill chain, the more distorted the timeline could be from the actual chronology of the attack. As can be seen in Figure 7, the distance of 14 (observed in th3bug, chafer, darkhydrus) is recorded as the minimum distance. 51 is the highest distance observed in Sofacy.

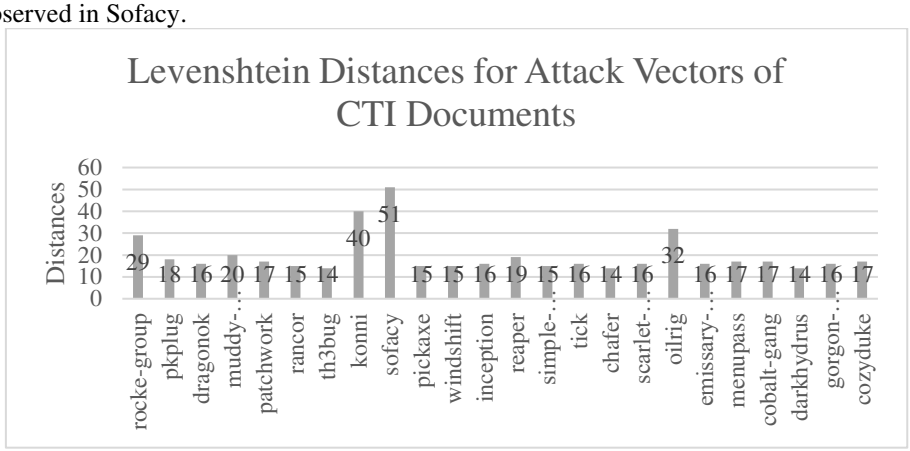

Figure 7. The distances from the unified kill chain order.

5 Conclusion

In this paper, we presented an approach to represent and visualize CTI data for further analysis in a timely manner. The proposed method constructs a timeline and a unified kill chain using indicators of varying "pain" level, in order to reach to the highest TTP level of the PoP. The unified kill chain can be very useful in the hands of a security operations analyst. An analyst can use this more practical approach of the unified kill chain as guide to perform various of tasks in his day to day activities. An analyst can use the proposed methodology for prevention. Given a CTI document such as the "muddy waters" from Figure 3 one can analyze and investigate the patterns. Having a timeline and the attack pattern codes in a single graph as shown in the Figure 2, the analyst has a quick reference guide on how the cyberattack has been deployed and which attack patterns codes have been deployed in each stage. Then the analyst can extract valuable information related to the attack pattern and use this information to create alerts, in a Security Information and Event Management (SIEM) in order to identify or even to prevent future attacks that follow similar patterns.

The proposed method may also be of use in incident response. The analyst can compare his incident flags and attack patterns, with the pre-analyzed datasets form the previous analyses and observe if any of the previous attack patterns match the current attack pattern. Consequently, the necessary counter measures and mitigation strategies can be decided.

As a future work, we will apply our approach into more populated datasets and try the methods with larger incidents. Another line of future work will be to apply this model to combinatorics on words for generating a pattern-matching approach for kill chain phased attack trees.

6 Acknowledgements

This work has received funding from the European Union's Horizon 2020 research and innovation program under the grant agreement no 830943 (ECHO).

7 References

1. ENISA: ENISA Threat Landscape Report 2018. (2019). https://doi.org/10.2824/622757.
2. Hutchins, E., Cloppert, M., Amin, R.: Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. 80–106 (2011).
3. Qiang, L., Zhengwei, J., Zeming, Y., Baoxu, L., Xin, W., Yunan, Z.: A Quality Evaluation Method of Cyber Threat Intelligence in User Perspective. Proc. - 17th IEEE Int. Conf. Trust. Secur. Priv. Comput. Commun. 12th IEEE Int. Conf. Big Data Sci. Eng. Trust. 2018. 269–276 (2018). https://doi.org/10.1109/TrustCom/BigDataSE.2018.00049.
4. Liao, X., Yuan, K., Wang, X., Li, Z., Xing, L., Beyah, R.: Acing the IOC Game : Toward Automatic Discovery and Analysis of Open-Source Cyber Threat Intelligence. 755–766 (2016).
5. Bianco, D.: Pyramid of Pain, http://detect-respond.blogspot.gr/2013/03/the-pyramid-of-pain.html, last accessed 2020/03/02.
6. Threat, I., Exchange, D.: Detect , SHARE , Protect Solutions for Improving Threat Data Exchange among CERTs. (2013).
7. ENISA: Exploring the opportunities and limitations o current Threat Intelligence Platforms. 42 (2017).
8. Rahayu, S.S., Robiah, Y.: Cyber threat intelligence – Issue and challenges Cyber Threat Intelligence – Issue and Challenges. 371–379 (2018). https://doi.org/10.11591/ijeecs.v10.i1.pp371-379.
9. Ponemon Institute: Third Annual Study on Exchanging Cyber Threat Intelligence: There Has to Be a Better Way. (2018).
10. Agency, E.U., Security, I.: Actionable Information for Security Incident Response. (2014).
11. Kompanek, A., Cc, C.: Evaluating Threat Intelligence Feeds FIRST Technical Colloquium for Threat Intelligence. (2016).
12. Faiella, M., Gonzalez-granadillo, G.: Enriching Threat Intelligence Platforms Capabilities. (2016).
13. Sillaber, C., Sauerwein, C., Mussmann, A., Breu, R.: Data Quality Challenges and Future Research Directions in Threat Intelligence Sharing Practice. 65–70.
14. Grispos, G., Glisson, W.B., Storer, T.: How Good is Your Data ? Investigating the Quality of Data Generated During Security Incident Response Investigations.
15. Sadiq, S.: Handbook of Data Quality. (2013).
16. Pols, P.: The Unified Kill Cahin. Cyber Secur. Acad. (2017).
17. Cichonski, P.: Computer Security Incident Handling Guide : Recommendations of the National Institute of Standards and Technology. NIST Spec. Publ. 800–61, 79 (2012). https://doi.org/10.6028/NIST.SP.800-61r2.
18. MITRE: A structured language for cyber threat intelligence, https://oasis-open.github.io/cti-documentation/, last accessed 2020/03/02.
19. Unit 42: Unit 42 Playbook Viewer, https://pan-unit42.github.io/playbook_viewer/, last accessed 2020/03/02.
20. MITRE: ATT&CK Framework, https://attack.mitre.org/, last accessed 2020/03/02.