Using CTI Data to Understand Real World Cyberattacks

Mauro Allegretta*, Giuseppe Siracusano†, Roberto Gonzalez†, Pelayo Vallina‡*, and Marco Gramaglia*
*Universidad Carlos III de Madrid, Spain

†NEC Laboratories, Germany

‡IMDEA Networks Institute, Spain

Abstract—**The forensic analysis of Cyber Threat Intelligence (CTI) data is of capital importance for businesses and enterprises to understand what has possibly gone wrong in a cybersecurity system. Moreover, the fast evolution of the techniques used by cybercriminals requires collaboration among multiple partners to provide efficient security mechanisms. STIX has emerged as the industrial standard to share CTI data in a structured format, allowing entities from over the world to exchange information to broaden the knowledge base in the area. In this work, we shed light on the type of information contained in these datasets shared among partners. We analyze a large real-world STIX dataset and identify trends for the reporting of CTI data. Then, we deep dive into two kinds of attack patterns found in the dataset: Command & Control and Malicious Software Download. We found the data is not only useful for forensic analysis but can also be used to improve the protection against new attacks.**

I. Introduction

Cyber-attacks require a thorough evaluation and analysis immediately after the incident to take mitigating actions such as quarantine network flows or sandbox applications. Despite the implementation of automated analysis to understand the root cause of the attacks and deploy the required software and hardware patches to solve the problem, this task still relies on manual validation. These manual efforts, which require very slow human inspections of text-based logs produced by the different sources, limit the capacity to prevent the threats by identifying common patterns in the attacks.

Network entities and organizations have implemented countermeasures to prevent these attacks by blocking content that has been previously identified as malicious or suspicious by other entities that have suffered such attacks. However, the lack of standardization on how they should report their incidents limits the capacity of other entities to take advantage of such previous experiences. To solve this limitation, different organizations have standardized in the last years the way of sharing Cyber Threat Intelligence (CTI) information with the STIX (Structured Threat Information Expression) format [1]. The STIX format represents incidents in entity-relationship graphs connecting different attack components meaningful for a specific threat. Initiatives like Hail a TAXII[1] or OpenCTI[2] make public this kind of forensic information in STIX format. However, only some private initiatives like the CyberTreatAlliance[3] use this information to improve cybersecurity solutions. Also, given the pervasiveness of modern wireless networks, cyber crimes are currently targeting also this kind of deployment. The most relevant and recent example is the one related to Viasat when a targeted cyber-attack managed to disrupt the service of the communication company [2], [3] in Ukraine (and some neighboring countries).

In this paper, we aim to study the capacity of these CTI datasets to implement solutions that improve global cybersecurity. We do this by performing an analysis of a large private STIX dataset, including reports for about 3M cyber incidents gathered in May 2021. We focus on two specific attack patterns present in the dataset: Malicious Software Download and Command & Control. We combine the CTI data with external data sources like VirusTotal(VT) [4] and Fortiguard [5] to discover how the global cyber security ecosystem reacted to the security incidents reported in the dataset. We also examine if those threats have been identified and neutralized several months after the first report.

Our results show that even several months after a threat has been reported in our dataset, it is still not identified and neutralized by different commercial solutions. Also, we find the possible causes behind this fact, as we observe malicious software relying on legitimate services such as Blogger, Discord, or GitHub to perform their activity. Altogether, the information included in the commonly shared STIX dataset is of great value to improve already existing solutions, and potentially create novel cyber threat detection techniques.

The paper is structured as follows: in Sec. II we introduce the STIX data bundles and review the current efforts in the analysis of cyber threat intelligence data, positioning our work. Then we focus on the analysis of our STIX dataset in Sec. III shedding light into two specific attack patterns: Malicious Software Download and Command & Control. Finally, we conclude the paper in Sec. IV.

[1] http://hailataxii.com/

[2] https://www.opencti.io/en/

[3] https://cyberthreatalliance.org/

[4] http://www.virustotal.com/

[5] http://www.fortiguard.com/

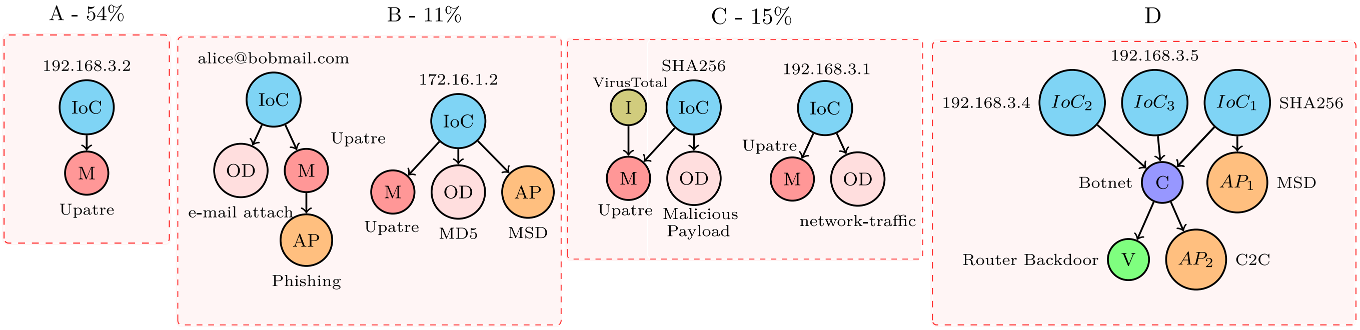

Fig. 1: Examples of STIX bundles. Percentages represent the popularity of each pattern in the dataset described in Section II-A

II. Context

In recent years, different organizations have standardized the way of sharing CTI information [4] with the STIX format. In STIX, cybersecurity incidents are represented as knowledge graphs that describe the relationship among different attack components meaningful for a specific attack kind.

A. The STIX format

In STIX, each cyber incident report is represented as a graph. The set of nodes representing different aspects of the attack and a set of edges describing the relations among the nodes are included in a bundle. More specifically, the information is categorized across different object types (called STIX Domain Objects, SDOs), The main item in a bundle is the Indicator of Compromise (IoC), which contains traces of the attack such as an IP address (in case of a network attack), an email (in case of, e.g., phishing), the hash fingerprint of a (malicious) software, etc.

We provide some examples of possible STIX bundles in Fig. 1. Bundles could be a very detailed security report with numerous nodes and edges, or just a triple with an IoC indicating a malicious entity as in Fig. 1 A. More complex layouts (Figures 1 B and C) can include other knowledge such as Attack Patterns (AP), Observed Data (OD), or external sources of information (I).

However, STIX can be used to build very complex compositions like in Fig. 1 D. In this example, the bundle describes the observation of a botnet attack campaign [5]. In this case, this information is further enriched by the attack patterns used Malicious Software Download and Command and Control (two use cases we further discuss in Section III) and the Vulnerability exploited, a router backdoor in this case. In summary, the STIX format is a valuable source and semantically rich representation of completely different cyber incidents.

The STIX dataset used in this paper contains information provided by more than 30 different cybersecurity vendors, with their own policies for disclosing CTI data. This heterogeneous origin causes the appearance of redundant nodes, especially when describing a trending attack.

To avoid redundancy, we start by merging different bundles into a single graph. First, we remove the syntactical duplicates, i.e. objects sharing the same Universal Unique Identifier (UUID). Afterward, following the STIX semantic equivalence guidelines[6], we identify the semantic duplicates. Those nodes represent the same object encoded with different UUIDs, e.g. *Malware* SDOs sharing the same "Name" field value but different UUIDs. All the duplicates are merged in a single node that inherits all the input and output relationships of its duplicates. With this methodology we build a connected graph.

B. State of the art

STIX datasets have already been leveraged in different ways. One prominent trend among CTI is to aggregate different sources provided in the form of textual reports or lists of indicators of compromise into a Semantic Database of Entities. For instance, the work in [6] proposes a Unified Cybersecurity Ontology (UCO). Then, several works build on similar concepts (i.e., ontologies) to retrieve knowledge graphs by feeding external CTI sources (including STIX providers) and applying semantic queries.

STIX knowledge graphs are usually employed as search engines from which one can derive assumptions that help and improve the work of a human expert. In [7], an external STIX dataset is used to derive a new database scheme and to extract well-defined security rules in standardized formats such as YARA and Snort. Finally, [8] constructs graphs by using UCO to extract entities from applications logs. So, STIX-based graphs are prominently used as databases to perform user-defined queries.

Still, these works rely on building ontologies starting from a structured database and integrating it with an external source of knowledge. This implies an additional phase of construction of the ontology and retrieval of the entities, often obtained by parsing plain text sources. This is the case of the works in [9] and [10], that propose the use of a heterogeneous information network instead of a canonical Resource Description Framework (RDF) triple extraction, to build the base graph. This setup is then used to perform a downstream task such as predicting the maliciousness of a domain that had interacted with network entities present in the graph

Hence, given the increasing use of the standard and the well-defined entity-relationship model, it is possible to avoid

[6] https://stix2.readthedocs.io/en/latest/guide/equivalence.html

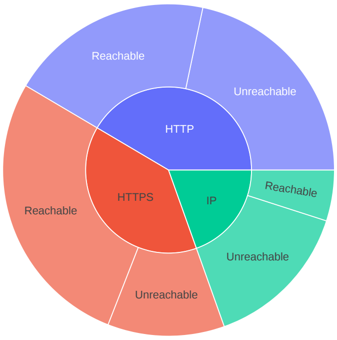

Fig. 2: Current status of the reported C&C attackers in our dataset.

the web semantic architecture, and build the graph by applying the rules of the standard, enriching where needed with custom external fields, and aggregating redundant information. In our work we only make use of STIX as the primary source of information, using it as a model to define the graph. This leads to much simpler construction and a well-defined set of entities and relationships. In the following, we discuss our approach that leverages Big Data techniques.

III. Case studies

We now analyze the utility of this data representation to understand common cyber threats that are unfortunately popular in our dataset: Command & Control (C&C) and Malicious Software Download.

A. *Command & Control Case Study*

C&C is a type of attack involving a malicious entity taking full control of a victim machine and executing arbitrary codes. It requires that an infected entity establishes a direct connection with a malicious one. Through the established channel, it is then possible to download malwares, retrieve command-line directives, etc. One way for establishing a connection, after the victim has been infected, is to use an existing, legitimate external Web service. We start by analyzing a set of IoCs directly linked with C&C, using the information available in the dataset. More specifically, we look at the Attack Pattern SDO, which contains the references to the Tactic Technique and Procedures (TTPs) defined in the MITRE ATT&CK Matrix (whose ID is TA0011)[7]

Hence, from the filtered dataset, we extract all the IP addresses and the URLs that point to C&C Attack Pattern, obtaining two lists: we then further split the latter into HTTP and HTTPS-based URLs. With this data, we can now check if those threats are still active even after several months. Using a Python script, we perform a HEAD request to each URL/IP in our dataset. From this point, we perform subsequent checks:

1) We contact each IP on port 80.
2) In case of failure, we contact the IP on port 443, checking the HTTPS certificate.

TABLE I: MSD targets identified by different data sources

Target	In Public CTI [%]	In Public CTI (6 months after) [%]	Blocklisted (VT) [%]	Blocklisted (Fortiguard) [%]
IP	54.33	70.01	38.1	11.83
FQDN	64.1	82.06	31.04	17.22
Total	**63.12**	**80.97**	**31.16**	**16.60**

3) We contact each HTTP/HTTPS URL, checking the certificate for the latter case.

In total, we collect answers from 6,761 URLs and 1,667 IPs. Results are reported in Figure 2, which shows the proportion of the different cases in our analysis. It can be noted that a large majority of the IPs and URLs are still reachable and working, serving potentially malicious software, even several months after the reported incident, and a large fraction of them are using a valid HTTPS certificate.

We dig more into these aspects and found that more than 6.000 URLs map to only 881 unique Fully Qualified Domain Names. Additionally, we found that 8.7% of those are also found among the Alexa Top 1M domains list [8].

Driven by this fact, we analyze them individually and discovered that many of them are belonging to very well-known web services. Specifically, we found that a substantial number of Command & Control malware leverages Google's well-known service Blogspot (more than 1,000 URLs point there), but also Discord, Dropbox, Google Docs, Github, and Bitbucket are present.

This aspect is a severe limitation for simple countermeasures against calls for automatized ways of detecting this behavior, as simply Fully Qualified Domain Name (FQDN) blocklisting may result into excessive limitations for the end users.

B. *Malicious Software Download Case Study*

We now focus on Malicious Software Download (MSD) as this type of attack is directly related to C&C. The attacker uses deceptive methods to cause a user or an automated process to download and install dangerous code coming from a malicious controlled source[9]. It is paramount to understand how the malicious software is downloaded to perform active countermeasures based on e.g. network monitoring. As a first step, we select from the dataset all the nodes pointing to MSD, which corresponds to 8% of the records. Then we analyze their distribution and reachability.

1) Download sources: All the download sources in our dataset are URLs: 59% of them use HTTPS protocol, while the remaining 41% use HTTP. Among them, 83.5% of the URLs contain an FQDN, while only 16.5% directly point to an IP address. By issuing DNS queries, we obtain the IP addresses associated with those FQDN: the 52K different FQDNs are compressed into 7,277 unique domains and 727 unique IPs. Hence, the same end host offers several download sources (i.e., URLs). By further analyzing the domains we observe that 68% of them were used to download only one malware, a percentage that drops to 17% for the domains that

[7] https://attack.mitre.org/tactics/TA0011/

[8] https://www.alexa.com/topsites.

[9] https://capec.mitre.org/data/definitions/185.html

hosted 2 malwares. This hints at an extended specialization of FQDNs. However, some FQDN exhibit a large degree of different malwares: in our dataset, Discord, Wetransfer, and Github were used as containers for tens of malicious software. Again, as discussed for the C&C case, blocklisting FQDN could not be a viable solution due to the extreme popularity of these web services.

2) Intelligence/Actions against Download Sources: The concerning amount of malwares that are still available for download even several months after they have been detected, gives us an intuition over the difficulties to remove this kind of content from the network. However, this problem may be solved if those resources are included in publicly accessible blocklists. With this purpose, we analyze the maturity of two publicly available categorization solutions for malwares, Virus Total (VT)[10] and Fortiguard[11]. Tab. I presents the amount of IPs and Domains that are known by the two commercial vendors. The first column indicates the FQDN/IPs already known by (any of the members of) VT at the moment in which they were inserted in our dataset. The second column indicates the number of FQDN/IPs known by VT 6 months after they appear in our dataset. The number of identified FQDN/Domains has grown from $54/64\%$ to $70/82\%$ since they appeared in the STIX dataset. So, 45% of the IP addresses and 36% of the domains were never marked as suspicious by any of the public CTI sources at the moment of the observation. Moreover, even 6 months after the observation, VT fails to mark as suspicious 30% of the IPs and 18% of the Domains. This gives an intuition on the possibility of using the STIX dataset to enrich public CTI information.

Finally, we analyze the FQDN/IPs that were eventually blocklisted by the two vendors we analyze (again, 6 months after appearing in the STIX dataset). In this case, the situation is even more concerning. Only 38% and 31% of the IPs and Domains, respectively, are included at least in one of the 69 blocklists monitored by VT.

Similar considerations apply also to Fortiguard. This tool provides tags to classify FQDN & IP such as “Business” or “Entertainment”, or reports about the security of a target (e.g., “Illegal or Unethical”). We hence consider a target malicious if it is tagged as “Malicious websites” or ”Phishing”. Even by considering two tags for this scenario, the amount of targets that are indicated in Fortiguard is well below 20 %.

This indicates a very conservative addition policy from all the blocklist providers. So, we believe that open circulation of the STIX dataset can be used to automatically enrich the publicly available blocklists.

IV. Conclusions

The use of forensic CTI is of key importance for the understanding and prevention of cyber threats. In this paper, we have analyzed a big STIX dataset containing reports for about 3M cyber events recorded in May 2021 to understand the potential of this tool and its graph representation. We then deep-dived into two popular attack patterns in our dataset: Command and Control and Malicious Software Download. For both of them, we discovered concerning aspects: even if several months passed since the incident, a large part of the malicious sources are still reachable and in many cases still delivering potentially harmful software. Also, we discovered that efficient blocklisting is difficult to achieve from the network side as most of the FQDN belongs to well-known web services that cannot be entirely made unavailable for e.g., a big enterprise.Thus, the main solution for countering this problem is still the usage of other threat databases that can be used directly by the clients to prevent connectivity towards malicious sites. Hence, we analyzed two very popular services (Virus Total and Fortiguard) showing that a still relevant fraction of malicious sources is not listed among their blocklisted entries. For all these reasons we believe that a systematic analysis of the graph obtained by STIX data (as we do in this paper) can be used to infer unwanted behaviour directly from network data.

Acknowledgement

The work of UC3M has been supported by the Spanish Ministry of Economic Affairs and Digital Transformation and the European Union-NextGenerationEU through the UNICO 5G I+D project 6G-RIEMANN. The work of NEC Laboratories Europe has been supported by the EU research project MARSAL (Grant Agreement 101017171).

References

[1] “STIX: Structured Threat Information Expression,” https://oasis-open.github.io/cti-documentation/stix/intro.

[2] Viasat, “KA-SAT Network cyber attack overview,” https://www.viasat.com/about/newsroom/blog/ka-sat-network-cyber-attack-overview/, 2022, [Online; accessed 09-Nov-2022].

[3] Sentinel LABS, “AcidRain — A Modem Wiper Rains Down on Europe,” https://www.sentinelone.com/labs/acidrain-a-modem-wiper-rains-down-on-europe/, 2022, [Online; accessed 09-Nov-2022].

[4] T. D. Wagner, K. Mahbub, E. Palomar, and A. E. Abdallah, “Cyber threat intelligence sharing: Survey and research directions,” *Computers & Security*, vol. 87, p. 101589, 2019.

[5] Antonakakis *et al.*, “Understanding the mirai botnet,” in *26th USENIX security symposium (USENIX Security 17)*, 2017, pp. 1093–1110.

[6] Z. Syed, A. Padia, T. Finin, L. Mathews, and A. Joshi, “UCO: A unified cybersecurity ontology,” in *Workshops at the thirtieth AAAI conference on artificial intelligence*, 2016.

[7] E. Kim, K. Kim, D. Shin, B. Jin, and H. Kim, “CyTIME: Cyber Threat Intelligence ManagEment Framework for Automatically Generating Security Rules,” in *Proceedings of the 13th International Conference on Future Internet Technologies*, ser. CFI 2018. New York, NY, USA: Association for Computing Machinery, 2018.

[8] A. Ekelhart, F. J. Ekaputra, and E. Kiesling, *The SLOGERT Framework for Automated Log Knowledge Graph Construction*, 05 2021, pp. 631–646.

[9] Y. Gao, X. Li, H. Peng, B. Fang, and S. Y. Philip, “HinCTI: A Cyber Threat Intelligence Modeling and Identification System Based on Heterogeneous Information Network,” *IEEE Transactions on Knowledge and Data Engineering*, pp. 1–1, 2020.

[10] J. Zhao *et al.*, “Cyber Threat Intelligence Modeling Based on Heterogeneous Graph Convolutional Network,” in *23rd International Symposium on Research in Attacks, Intrusions and Defenses (RAID 2020)*. San Sebastian: USENIX Association, Oct. 2020, pp. 241–256.

[10] www.virustotal.com

[11] www.fortiguard.com