Sharing cyber threat intelligence: Does it really help?

Beomjin Jin*, Eunsoo Kim*, Hyunwoo Lee†, Elisa Bertino‡, Doowon Kim§ and Hyoungshick Kim*
*Department of Electrical and Computer Engineering, Sungkyunkwan University
†Department of Energy Engineering, Korea Institute of Energy Technology (KENTECH)
‡Department of Computer Science, Purdue University
§Department of Electrical Engineering and Computer Science, University of Tennessee
*{jinbumjin, eskim86, hyoung}@skku.edu, †hwlee@kentech.ac.kr, ‡bertino@purdue.edu, §doowon@utk.edu

***Abstract*—The sharing of Cyber Threat Intelligence (CTI) across organizations is gaining traction, as it can automate threat analysis and improve security awareness. However, limited empirical studies exist on the prevalent types of cybersecurity threat data and their effectiveness in mitigating cyber attacks. We propose a framework named CTI-Lense to collect and analyze the volume, timeliness, coverage, and quality of Structured Threat Information eXpression (STIX) data, a *de facto* standard CTI format, from a list of publicly available CTI sources. We collected about 6 million STIX data objects from October 31, 2014 to April 10, 2023 from ten data sources and analyzed their characteristics. Our analysis reveals that STIX data sharing has steadily increased in recent years, but the volume of STIX data shared is still relatively low to cover all cyber threats. Additionally, only a few types of threat data objects have been shared, with malware signatures and URLs accounting for more than 90% of the collected data. While URLs are usually shared promptly, with about 72% of URLs shared earlier than or on the same day as VirusTotal, the sharing of malware signatures is significantly slower. Furthermore, we found that 19% of the *Threat actor* data contained incorrect information, and only 0.09% of the *Indicator* data provided security rules to detect cyber attacks. Based on our findings, we recommend practical considerations for effective and scalable STIX data sharing among organizations.**

I. Introduction

Cyber attacks have increased in number, and their risks are becoming more severe [1]. A recent report [20] reveals a 435% increase in the number of ransomware attacks from 2019 to 2020. It is also reported that 68% of industry practitioners felt that cybersecurity risks would threaten their business [2] and 56% of utilities were targeted by cyber attacks [65]. Not only is the scale of cyber attacks growing, but the costs incurred by victims are also rising. For example, the estimated cost of the Solarwinds attack is about $100 billion [54].

Cyber Threat Intelligence (CTI) has emerged as an indispensable component of standard security procedures, assisting organizations in risk prioritization, timely detection, mitigation, and containment of attacks [25]. A SANS report [18] indicates that nearly 60% of organizations are already utilizing CTI in their security practices, and 25% plan to integrate it into their security operations. The report also unveils that nearly 47% of them have dedicated CTI teams. Sharing CTI data across organizations is instrumental in countering sophisticated attacks such as Advanced Persistent Threats (APTs). Comprehensive CTI data can be leveraged to identify indicators linked to one or more stages (*e.g.*, reconnaissance, initial compromise, lateral movement, and data exfiltration) of these attacks [47]. For instance, ransomware threats like Locky [24] and WannaCry [44] employ a similar method of infection–emails containing malicious attachments or links. A shared characteristic between these attacks is the use of the Tor network for their command and control communication. Once a ransomware sample infiltrates a system, it propagates laterally, aiming to compromise others in the network. The kill chains for these attacks, which occurred almost a year apart, showed significant overlap; knowledge of the former attack could have facilitated preparations for the subsequent one.

Several standards are available to represent and share threat information in a structured manner, such as Structured Threat Information eXpression (STIX) [57], Cyber Observable eXpression (CybOX) [11], and Malware Information Sharing Platforms (MISP) [13]. Sharing CTI data in a structured format makes it easier to automate data processing and analysis. STIX has become the *de facto* standard [19], [3] for representing CTI data because it was expressly designed to represent various levels of CTI information in a machine- and human-readable format [56]. Furthermore, a standard protocol, Trusted Automated eXchange of Indicator Information (TAXII) [33], facilitates the sharing of STIX data across organizations. Therefore, we focus on STIX analysis.

Despite the availability of STIX, companies perceive that vital CTI (*e.g.*, campaigns, motivations, and TTPs) is not effectively disseminated in practice [25]. A recent survey found that many organizations value high-level CTI (*e.g.*, TTPs, threat actors, security rules, and APT campaigns) more than simple indicators of compromise (IoCs), such as malware hashes, IP addresses, and URLs, for developing effective defenses against sophisticated attacks [10]. However, despite the recognized importance of high-level CTI, little research has been done to understand how high-level CTI is shared in the wild. Previous research in open CTI has predominantly focused on a limited range of basic IoC data types [26], [27], [17], [73].

To address the lack of knowledge on the STIX data sharing landscape, we collect STIX data from all publicly available open CTI sources, enabling comprehensive large-scale anal-

Network and Distributed System Security (NDSS) Symposium 2024
26 February - 1 March 2024, San Diego, CA, USA
ISBN 1-891562-93-2
https://dx.doi.org/10.14722/ndss.2024.24228
www.ndss-symposium.org

ysis. This analysis is essential for enhancing the sharing of advanced cybersecurity threat data, as it will help identify key challenges inhibiting organizations from effectively sharing such data.

To this end, we develop a framework named CTI-Lense[1] that collates STIX data from a set of open CTI sources and systematically analyzes the collected data. The analysis evaluates the data on four measurements: volume, timeliness, coverage, and quality on a large scale. CTI-Lense facilitates more efficient and accurate analysis of the state of open CTI data sharing. To demonstrate the potential of CTI-Lense, we collect approximately 6 million STIX data points from October 31, 2014 to April 10, 2023 (a span of nine years) across 10 data sources and conduct a comprehensive analysis of these data. We summarize our findings as follows.

Volume. STIX data sharing has steadily increased in recent years, but the volume of STIX data shared is still relatively low to cover all the cyber threats arising. Our investigation reveals that security service providers generate and share an average of only 2,063 unique STIX objects per day. This quantity seems significantly inadequate to handle the daily influx of new malware samples, malicious IPs, and URLs, particularly considering that the AV-TEST Institute records over 450,000 new malware samples and potentially unwanted applications each day [6]. Furthermore, our research found that 37.89% of STIX objects generated even by a single service provider are duplicated, with `JamesBrine` exhibiting the highest duplication rate at 68.74%. However, the percentage of duplicated data across different service providers remains relatively low at 4.10%. Based on these findings, consumers of STIX data should contemplate implementing additional deduplication procedures and collecting STIX data from various sources to enhance the heterogeneity of STIX data.

Timeliness. We find that URL objects are typically disseminated promptly, with approximately 72% and 88% of URLs shared either before or on the same day as their appearance on VirusTotal and HybridAnalysis, respectively. This prompt sharing of URLs within the first 24 hours could potentially prevent a considerable portion of malicious domains, considering their typically short lifespans [23]. For example, many spam domains are active for merely a day, aiming to evade detection and avoid inclusion in blocklists [72]. On the contrary, the sharing of file-type threat objects (*e.g.*, malware signatures) is notably slower than that of VirusTotal. This observation indicates potential inefficacy in preventing malware attacks, especially considering that 54% of malware samples are active for only 24 hours [48]. In addition, we note that STIX data is commonly shared two to four days after reporting security incidents. Our data show weak evidence of *causality* [51] from security incidents reported in sources like Malpedia and security news websites to the sharing of STIX data in our collected dataset, with $p < 0.05$ for the 2–4 and 2–12 days time-lag, respectively. However, we do not find any evidence of causality between the publication of CVEs and STIX data.

Coverage. The STIX standard [57], [9] is designed to represent a wide range of sophisticated cybersecurity threat information, such as APTs, threat actors, and their relationships. However, our analysis of shared STIX data shows that only a limited number of STIX data types are used. Most STIX data represent simple IoCs, such as malware signatures and URLs. This suggests that the STIX standard is not being used to its full potential. We also found that many of the objects and attributes specified in the STIX standard are not used in practice. For example, only 0.09% of indicators contain security rules that can be used for detecting cyber attacks in an automated manner. This suggests that more guidance is needed on how to use the STIX standard to represent complex threat information.

Quality. We evaluate the quality of the values contained in the STIX data from two aspects: correctness and completeness. Correctness refers to whether the STIX data are appropriately used with the right objects, attributes, and values as intended in the STIX standard. For example, we found that 19% of the STIX data have incorrect *Threat actor* values. Completeness refers to whether the STIX data accurately contain valid threat information. For example, we found that over 50% of the *Indicator* objects represent their information in narrative forms as existing threat reports. This suggests that additional manual efforts are required to process STIX objects and extract relevant threat information.

II. Background and Motivation

A. What are CTI and STIX?

Cyber Threat Intelligence (CTI). CTI is evidence-based knowledge about cybersecurity threats that supports security analysts in making informed decisions. It provides information about threats at all levels, from low-level indicators such as basic IoCs (*e.g.*, malware hashes and IP addresses) to high-level indicators (*e.g.*, TTPs). CTI data is often disseminated in unstructured forms, such as narrative reports, blog posts, or vendor documents. This requires significant preprocessing, including deduplication and structurization, before it can be used to make any decisions on threats in an automated method [10]. Structured CTIs, such as STIX and MISP, address this challenge by providing a machine-readable format for CTI data. This makes it more efficient to manage and analyze CTI.

Structured Threat Information eXpression (STIX). The heterogeneity of CTI specifications, each with its own structure and data expression method, challenges organizations when sharing intelligence. To address this challenge, the US Department of Homeland Security (DHS) and MITRE[2] collaborated to develop STIX, a standard that provides a structured language for expressing threat information.

STIX ecosystem. Figure 1 illustrates the entities within a STIX ecosystem. A standardization organization (*e.g.*, OASIS) publishes the STIX specification (❶). A producer (*e.g.*, a security analyst) analyzes cyber threats and generates STIX data according to the STIX standard (❷). The producer then shares the STIX data via platforms (*e.g.*, STIX data repositories or TAXII servers) managed by a service provider like MITRE or AlienVault OTX (❸). A consumer then fetches the STIX data from such a platform using a standard protocol such as Git or TAXII (❹). Finally, the consumer employs this data to analyze security incidents or minimize the risk of cyber

[1] https://github.com/SKKU-SecLab/CTI_Lense

[2] MITRE is a non-profit R&D organization supporting various US government agencies in aviation, defense, healthcare, homeland security, and cybersecurity domains.

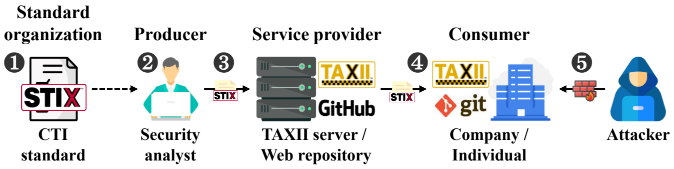


Fig. 1: Ecosystem of STIX. A standard organization is responsible for publishing STIX specifications. Producers generate STIX data and upload them to a platform managed by a dedicated provider. Consumers then access and retrieve this STIX data using an industry-standard protocol such as TAXII.

```
<stix:STIX_Package ...>
  <stix:Indicator
    id="opensource:indicator-..."
    timestamp="2014-11-05T03:05:02.125545+00:00"
    version="2.1.1">
    <indicator:Title>
      ... Reported CnC ...
    </indicator:Title>
    <indicator:Type>IP Watchlist</indicator:Type>
    <indicator:Description>
      SNORT Rule ...
    </indicator:Description>
    <indicator:Observable idref="...">
      <cyboxCommon:address_value>
        78.46.33.91
      </cyboxCommon:address_value>
    </indicator:Observable>
    <indicator:Indicated_TTP>
      <ttp:Name> ET.Evil </ttp:Name>
    </indicator:Indicated_TTP>
    <indicator:Test_Mechanisms>
      <snortTM:Rule>
        alert tcp HOME_NET any ...
      </snortTM:Rule>
    </indicator:Test_Mechanisms>
    <indicator:Producer>
      <stixCommon:Name>
        rules.emergingthreats.net
      </stixCommon:Name>
    </indicator:Producer>
  </stix:Indicator>
</stix:STIX_Package>
```

Fig. 2: Example of a STIX 1 document. The STIX data consists of objects (*e.g.*, stix:Indicator) and their attributes (*e.g.*, indicator:Title) written in an XML format (STIX 1) or a JSON format (STIX 2).

threats, for instance, by extracting indicators from the STIX data and implementing them in firewall rules (❺). In a real-world STIX ecosystem, there are typically multiple service providers, producers, and consumers. Producers and consumers can be enterprises, financial institutions, or even individuals.

STIX objects and attributes. STIX data comprise various types of cyber threat information such as observables, indicators, incidents, Tactics, Techniques and Procedures (TTPs), exploit targets, courses of action, campaigns, and threat actors. Information within STIX data is depicted as *objects* and their corresponding *attributes*. Figure 2 shows an example of STIX 1 data representing indicator information, which includes the IP address of the command and control (C&C) server related to `ET.Evil` malware, as well as the Snort rule for detecting such network traffic. In this figure, a threat indicator is expressed as an *Indicator* object (*i.e.*, `<stix:Indicator>`. An object's details, such as *Title* and *Type*, are represented as attributes (*i.e.*, `<indicator:Title>` and `<indicator:Type>`).

STIX 1 and STIX 2. STIX 1, introduced by MITRE in 2012, uses an XML format to describe threats and supports nested structures (*i.e.*, objects within objects). However, these nested structures often render STIX data intricate and challenging to parse. Additionally, STIX 1 allows for multiple representations of the same threat, leading to the existence of two STIX objects that describe the same target with different structures. To overcome the limitations of STIX 1, STIX 2 was standardized by OASIS in 2017. The main distinction between STIX 1 and STIX 2 is in their structures: STIX 2 uses the JSON format, making it more lightweight and easier for programmers to work with, and it favors a more flat and relational data model over deeply nested structures. Although the CTI ecosystem is gradually transitioning to STIX 2, many service providers continue to share STIX 1 data. Hence, it is crucial to analyze STIX 1 data as well. To perform analysis, we collect and examine both versions of the STIX data.

B. Trusted Automated eXchange of Indicator Information

The Trusted Automated Exchange of Intelligence Information (TAXII) [33] is a protocol for exchanging CTI data over HTTPS. TAXII defines APIs, enabling producers (or publishers) to share their CTI data with consumers (or subscribers). TAXII outlines two primary services to support different types of sharing: (1) a TAXII collection and (2) a TAXII channel. A TAXII collection is used by CTI producers to store CTI data on a data repository managed by a service provider. A TAXII channel is employed to exchange CTI data between TAXII servers (*i.e.*, publisher pushing CTI data) and TAXII clients (*i.e.*, consumer requesting data to obtain CTI data). A server typically manages several data repositories through these two services to publicly share STIX data with consumers. However, TAXII does not support backward compatibility with regard to STIX 1 and STIX 2. In other words, a TAXII server cannot share both versions of STIX. As we deal with both STIX 1 and 2 data in our analysis, we collect our data from open TAXII 1 and 2 servers, respectively.

C. Why Do We Focus on STIX?

Despite the need for advanced threat intelligence (*e.g.*, TTPs, threat actors, security rules, and APT campaigns), there is a lack of research on the sharing of structured CTI that can effectively represent such information. Therefore, we focus on STIX in our CTI analysis because it is the *de facto* standard for structured CTIs [19], [3] and is widely used in cybersecurity due to its effectiveness in representing complex threat data [18], [53]. STIX is used in various domains, including security policy development, threat hunting, and risk monitoring. For example, Syam et al. [5] propose a trusted response management ecosystem where STIX serves as a unified language for threat intelligence providers to seamlessly use CTIs for defensive actions. STIX also plays a crucial role in formalizing ontology for CTI, enabling connections between security events and CTIs, and allowing the inference of new knowledge about potential threats [40]. Prominent entities in the cybersecurity sector, such as SANS [31] and

Intel [30], incorporate STIX indicators into their threat-hunting methodologies. These indicators, such as IP addresses, domain names, and host artifacts, serve as the foundation for formulating security rules for network monitoring. Automated alerts generated from threat-hunting platforms can be exported via TAXII in a STIX-compatible format, which can then be integrated into security information and event management (SIEM) systems.

III. Research Question & Methodology

This section presents the research questions we aim to answer, the design of our framework (CTI-Lense), and the dataset we employ to answer these research questions.

A. Research Questions

Our research has two main goals. *First*, we aim to examine the volume, timeliness, and coverage of STIX data disseminated in the public domain to gain insights into its usage. *Second*, we aim to evaluate the quality of STIX data in accurately representing cybersecurity threat information, including malware, threat actors, and TTPs. To achieve these goals, we raise the following research questions:

RQ1 (Volume): What is the extent of STIX data that is being generated and shared publicly? Our objective is to quantify the amount of STIX data publicly released from open sources. Specifically, we examine the count of unique STIX objects shared daily. This analysis provides insights into the quantity of shared STIX data to determine if the volume of publicly shared STIX data is sufficient to cover a massive number of new cyber threats. Next, we are also interested in how many duplicated data are in the collected dataset to see the substantial scale of the shared STIX data. A lot of redundant data results in a smaller coverage of the target threats that STIX describes. Furthermore, it means that security applications that rely on STIX objects should perform the unnecessary deduplication process. Lastly, we investigate the number of STIX data over time for multiple sources. It aims to find any trend in STIX adoption. We also focus on the existence of sources where data is consistently shared.

RQ2 (Timeliness): How promptly is STIX data shared following a cyber threat discovery? Our objective is to assess the time interval from when a threat is initially detected to when a corresponding STIX object is publicly released since the timeliness of threat intelligence plays a pivotal role in cybersecurity. This analysis provides a clearer perspective on how efficiently threat intelligence sharing can act as a deterrent against cyber attacks.

RQ3 (Coverage): To what extent does publicly shared STIX data cover a wide range of cybersecurity threats? Our objective is to evaluate the coverage of threat information disseminated via the STIX standard. STIX defines a wide range of objects and attributes to represent different aspects of cybersecurity threats, from basic IoCs (*e.g.*, malicious IPs and file hashes) to sophisticated TTPs (*e.g.*, threat actors and malware categories). We analyze the distribution of STIX objects and attributes in a large dataset of publicly available STIX data. This analysis provides a deeper understanding of the extent to which the STIX standard is being used to represent a wide range of cybersecurity threats.

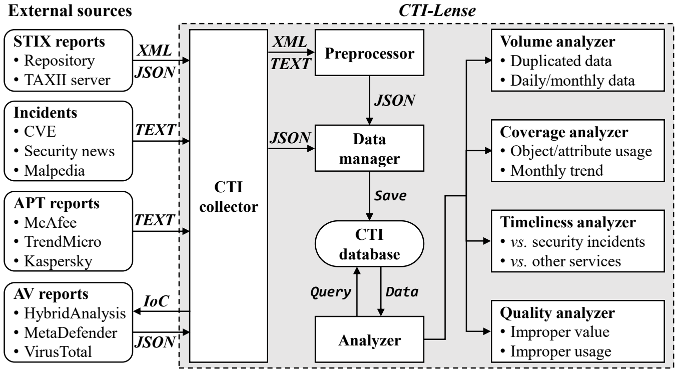


Fig. 3: Overview of CTI-Lense.

TABLE I: CTI source information used to evaluate STIX data in CTI-Lense.

CTI data	Format	Sources	Contained info
STIX	XML, JSON	• TAXII • Repository	• STIX 1 obj/attr • STIX 2 obj/attr
Incidents	TEXT	• CVE • Malpedia • Security news	• Title • Description • Publication time
APT report	TEXT	• Kaspersky • TendMicro • McAfee • ESET • Fortinet	• Malicious IoCs • Threat actor • Malware type
AV report	JSON	• VirusTotal • MetaDefender • HybridAnalysis	• AV scan results • First submission time

RQ4 (Quality): What is the quality of STIX data in terms of its correctness and completeness in representing cyber threats? Our objective is to evaluate the quality of STIX data by assessing its correctness and completeness in representing crucial cybersecurity information. We investigate if the objects and attributes within the STIX data are used accurately according to the STIX standard, and if producers assign valid values correctly. This analysis provides insights into the reliability of the disseminated STIX data.

B. Overview of Our Framework (CTI-Lense)

To answer our research questions, we developed CTI-Lense, a framework that aggregates CTI data from multiple open sources and analyzes the compiled data to assess its volume, timeliness, coverage, and quality. We use CTI-Lense to analyze STIX data. Figure 3 illustrates the overall architecture of CTI-Lense, which collects CTI data (particularly STIX data) from various sources to better understand the STIX ecosystem. The format, full list of sources, and information for each CTI data are listed in Table I.

We primarily analyze STIX as our CTI dataset, evaluating volume, timeliness, coverage, and quality based on its objects and attributes. The "incident" dataset is a narrative text dataset sourced from Malpedia, security news websites, and CVEs.

It contains the title, description, and publication time, which we use to perform causality tests to measure timeliness. The "APT report" dataset is another narrative text dataset reported by security experts from prominent companies, such as Kaspersky. This dataset provides information on malicious IoCs, threat actors, and malware types and helps to evaluate the quality of the STIX data. For text-based CTI datasets like "incident" and "APT reports," we transform them into JSON format, incorporating both raw text and essential details for comparison with STIX data. Finally, the "AV report" dataset is a JSON-formatted CTI dataset collected from three widely used scanning services: VirusTotal, MetaDefender, and HybridAnalysis. It contains anti-malware scanning results and IoC submission times, which we use to analyze the timeliness and quality of the STIX data.

Volume. We assess the volume and duplication of STIX data to understand the amount and quality of information. First, we count the number of STIX data objects shared daily and by source. Then, we measure the proportion of duplicated STIX data within and across sources by comparing objects based on unique attribute values. Finally, we measure the number of monthly shared STIX data for each source (see Section IV).

Timeliness. We assess the timeliness of STIX data by comparing its public release time with the timing of security incidents and its registration time to popular scanning services such as VirusTotal [71]. This allows us to determine how quickly STIX data is shared compared with commercial services. First, we investigate how long it takes for STIX data to be publicly released after a security incident occurs. We collect information about security incidents from three sources: Malpedia [16], security news websites [35], [29], and CVEs [7]. We also perform a Granger causality analysis [62] to determine whether the occurrence of security incidents causes more sharing of STIX data objects. Second, we compare the public release time of STIX data with its registration time to popular scanning services such as VirusTotal [71], HybridAnalysis [12], and MetaDefender [46] (see Section V).

Coverage. We assess the threat type coverage in STIX data by measuring the number of objects and attributes producers use to represent cybersecurity information. We count the objects and attributes in our collected dataset and compare them to the total specified in the STIX standard. We analyze object and attribute usage statistics to see which types are used, and how these have changed over time. This allows us to understand the threat information types used in STIX data and if the STIX standard is utilized to its full potential (see Section VI).

Quality. We assess the quality of STIX data by examining *improper values* and *improper usage*. *Improper values* refer to instances of incorrect values that violate the correctness of STIX. We evaluate *improper values* by investigating whether the information associated with objects or attributes is correct and accurate. For keyword type values (*e.g.*, threat actors and malware categories), we obtain valid keywords as correct information from diverse trustworthy data sources. We consider several valid keywords because several producers use their own unique attribute values. For example, some producers use "Lazarus Group," while others use "Guardians of Peace" or "Whois Team." We check whether a given keyword matches one of these valid keywords. We also check whether indicator objects (*e.g.*, malware hash, IP, and URL) refer to malicious entities by comparing each indicator object's value with the scanning reports obtained from online scanning services. We use three prominent online scanning services, VirusTotal [71], HybridAnalysis [12], and MetaDefender [46], to determine whether a given data is malicious using a threshold-based method [64]. *Improper usage* refers to cases where values exist in an object but are not assigned to the precise attributes, which violates the completeness of STIX. For example, a malware family name is described in the `description` attribute in a *Report* object rather than the `name` attribute in a *Malware* object. We evaluate *improper usage* by investigating whether values are incorrectly assigned to improper objects or attributes when they exist (see Section VII).

TABLE II: List of STIX data sources (TAXII servers (T) and repositories (R)), the number of total and unique objects collected from each source, and the proportion of duplicated (dup.) objects in each STIX data source within the same source (In dup.) and across different sources (Out dup.).

STIX sources	Total	Unique	In dup.	Out dup.
STIX 1				
Hail a TAXII (T)	3,820,542	1,900,237	50.71%	0.76%
AlienVault OTX (T)	2,697,680	1,647,509	38.93%	1.16%
IBM X-Force Exchange (T)	628,738	273,274	46.36%	20.32%
PickupSTIX (T)	100,859	73,575	18.13%	6.86%
STIX 2				
AlienVault OTX (T)	1,739,017	1,657,442	3.02%	2.00%
JamesBrine (R)	658,282	205,776	68.74%	0.00%
DigitalSide (R)	298,598	198,439	33.52%	7.78%
Cyware (T)	263,633	228,782	11.94%	2.51%
IBM X-Force Exchange (T)	122,717	119,611	1.25%	2.54%
Unit42 (T)	38,201	33,379	7.03%	13.29%
MITRE ATT&CK (R)	16,936	17,042	0.05%	1.73%
Limo from Anomali (T)	7,170	6,492	0.06%	12.68%
PickupSTIX (T)	516	507	1.74%	0.00%

C. Dataset

To conduct our analysis, we compiled a dataset of STIX data from seven publicly available TAXII servers and three repositories, as outlined in Table II. The observation period spanned from October 31, 2014 to April 10, 2023, during which we gathered a total of 6,362,065 objects (3,894,595 from STIX 1 and 2,467,470 from STIX 2). At first glance, the quantity of STIX data instances we gathered may appear to be smaller than the number of reported IoCs in the previous study [70]. However, a direct comparison between these two statistics is not straightforward. The IoCs reported in [70] contain all IoCs provided by *producers*, a significant portion of which would be duplicates. In contrast, our study focuses exclusively on refined and actively shared CTI data provided by *service providers*, as illustrated in Figure 1.

We selected seven TAXII servers: `AlienVault OTX`, `Hail a TAXII`, `IBM X-Force Exchange`, `Cyware`, `PickupSTIX`, `Unit42`, and `Limo from Anomali`; along with three public repositories: `JamesBrine`, `DigitalSide`, and `MitreAttack`. To aggregate as many public STIX data sources as possible, we refer to the services listed on the official STIX website [43] and data sources referenced in academic literature [3] and public communities [66], [42]. There are 84 STIX-related services in total, and we finally select 10 services among them.

The reasons why we remove 74 services are because 23 of them are inaccessible and 19 of them are confined to closed communities or necessitate payment. The rest 32 services are ones that only leverage STIX (*e.g.*, SIEM) but do not share their STIX data through a TAXII server or a repository.

To facilitate effective analysis of STIX objects, we store the content of the collected STIX objects in our internal MongoDB database. To handle two major STIX versions – STIX 1.x and STIX 2.x, which differ in structure (flat vs. nested) and format (XML vs. JSON), we convert STIX objects that are formatted in XML (*i.e.*, STIX 1) into JSON-encoded ones using the `python-stix`[3] library.

STIX allows for references within objects. However, these references can have varying styles, complicating processing. For instance, a STIX 1 object from `AlienVault` incorporates an *Observable* object as one of the attributes within an *Indicator* object. Conversely, `Hail a TAXII` generates separate *Observable* objects and references them within an *Indicator* object using the *id* attribute from the Observable objects. We tackle this challenge by ensuring all *Observable* objects exist as independent entities with unique *id*s when we store STIX objects in our database.

The final step in our data collection process is deduplicating the STIX objects. This is another challenging task because different collections can have different STIX objects representing the same threat information, even with different identifiers. For example, two STIX objects might both represent a malware sample, but they might have different identifiers because they were collected from different sources. To address this challenge, we follow a two-step deduplication process. First, we remove objects with identical identifiers that were collected from the same data source. This ensures that we do not have multiple copies of the same object from the same source. Second, we eliminate duplicate objects based on unique attribute values. For example, we might compare the hash values of malware samples or the IP addresses of malicious domains. This ensures that we do not have multiple copies of the same object from different sources. After deduplication, our dataset comprises 6,362,065 objects, which were deduplicated from 10,392,889 objects.

IV. Volume of STIX Data in the Wild

This section provides several statistical measures of publicly shared STIX data using CTI-Lense. We first quantify the volume of STIX data by measuring the number of publicly shared STIX objects, analyzing in more detail with regard to STIX version and data source. Second, we analyze the extent of duplicated data within the dataset. Finally, we measure the number of monthly shared STIX data for each source to analyze the volume trend over time.

Number of STIX objects shared. In our dataset, we find that the first shared STIX data appeared on October 31, 2014. In total, 10,392,889 STIX objects were shared until April 10, 2023, equating to an average daily share of 3,371 STIX objects. Of the total number of shared STIX objects, 7,247,819 (69.74%) are in STIX 1 format and 3,145,070 (30.26%) are in STIX 2 format.

For STIX 1 data, `Hail a TAXII` emerged as the dominant source, contributing 3,820,542 objects, which represent 52.70% of the total 7,249,819 objects. The second largest contributor was `AlienVault OTX`, with its share amounting to 2,697,680 objects or 37.21%. `IBM X-Force Exchange` and `PickupSTIX` together accounted for approximately 10% of the shared data, with contributions of 8.67% and 1.39%, respectively. For STIX 2 data, 2,171,254 objects (69.04%) were shared via TAXII, and external repositories contributed 973,816 objects (30.96%). Among the entities that share STIX 2 data over TAXII, `AlienVault OTX` was the most prolific, contributing 1,739,017 objects, which represent more than 80% of the shared data.

For all STIX objects in our dataset, we observed that more than 9,419,073 objects (90.63%) were shared through TAXII and 973,816 objects (9.37%) were shared from external repositories. The top 3 most frequent data sources over TAXII are `AlienVault OTX`, `Hail a TAXII`, and `IBM X-Force Exchange`, sharing 4,436,697 (47.10%), 3,820,542 (40.56%), and 751,455 (7.98%) of the total shared STIX objects, respectively. For external repositories, `JamesBrine` was the most prevalent, sharing 658,282 objects (67.60%), followed by `DigitalSide`, sharing 298,598 objects (30.66%), and `MITRE ATT&CK`, sharing 16,936 objects (1.74%).

Data duplication. Table II shows the number of STIX objects shared by each data source from the moment we first observed their sharing activity to the end of our data collection period.

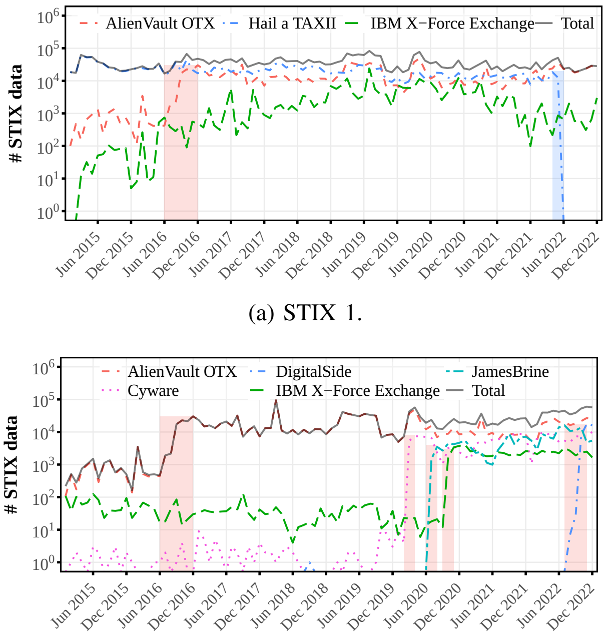
Our analysis of unique objects shared by data sources in both the STIX 1 and STIX 2 data reveals interesting trends. Contrary to what one might expect, the volume of shared objects does not directly correlate with the number of duplicates across various data sources. For instance, `Hail a TAXII`, which shares the highest number of objects, has the lowest proportion of duplicates at only 0.76%. This is followed by `AlienVault OTX` at 1.16%, `PickupSTIX` at 6.86%, and `IBM X-Force Exchange` at 20.32%. Similarly, in STIX 2, each data source possesses a significant number of uniquely shared STIX objects not found in other data sources. These results are consistent with previous work showing that most data in threat intelligence feeds are unique [41], [68], [26].

However, we observe a significant amount of duplication within individual data sources. For instance, `Hail a TAXII` shows a 50.71% duplication rate in its shared STIX 1 objects. Other sources present similar trends: `IBM X-Force Exchange` contains 46.36% duplicates, `AlienVault OTX` contains 38.93%, and `PickupSTIX` contains 18.13%. Considering that `Hail a TAXII` and `AlienVault OTX` together account for 89.91% of all shared STIX 1 objects, the high rate of duplication within these sources raises concerns about data redundancy.

In contrast, the rate of duplicates in STIX 2 data shared via TAXII is relatively lower, but the concentration of duplicates is higher for data shared through external repositories. Among these, `JamesBrine` shared the highest number of STIX 2 objects (658,282), with 68.74% of which are duplicates. `DigitalSide`, the second most prevalent repository, shared 298,598 objects, 33.52% of which are duplicates.

Overall, 37.89% of the STIX objects generated by a single source are redundant, with `AlienVault OTX` exhibiting the

[3] https://github.com/STIXProject/python-stix



(a) STIX 1.

(b) STIX 2.

Fig. 4: Monthly number of shared STIX data. The red zones indicate periods of rapid increase in STIX data volume, while the blue zone indicates a period of rapid decrease in volume.

highest duplication rate at 68.74%. Of the total 10,392,889 shared STIX objects, only 6,362,056 (61.22%) are unique across all data sources. Consequently, a daily average of just 2,063 unique objects is publicly shared. Given the daily influx of new malware samples, IPs, and URLs, this number seems insufficient, especially considering AV-TEST Institute's records indicating that over 450,000 new malware samples and potentially unwanted applications emerge each day [6].

STIX data sharing volume over time. Figure 4 shows how many STIX data (1 and 2) have been shared by each source during our observation period, excluding sources with less than 100,000 unique data for better presentation.

STIX data sharing has increased in recent years. As shown in Figure 4a, over 10,000 STIX data points have been shared on `Hail a TAXII` in most months since January 2015. `AlienVault OTX` experienced a sharp rise in shared STIX 1 data starting in June 2016, with nearly 10,000 STIX data points being shared in most months after that. `IBM X-Force Exchange` has consistently shared over 1,000 STIX data points monthly since March 2018. These three sources significantly contributed to the dataset and consistently shared STIX data until `Hail a TAXII` stopped sharing in June 2022. As shown in Figure 4b, for STIX 2, `Cyware`, `DigitalSide`, `IBM X-Force Exchange`, and `JamesBrine` have actively shared data since the release of the latest STIX 2.1 version in March 2020 [9]. Because STIX 2 was released in July 2017, there should technically be no data created before this date. We have found cases where the creation time of STIX 2 data is earlier than July 2017. This occurred because some STIX 1 data was converted to STIX 2 after July 2017, resulting in the retention of the original creation time in the new format. We verified this by extracting indicators from both STIX 1 and STIX 2 data that shared identical hash values, confirming that STIX 1 data was utilized in the generation of STIX 2 data. Overall, the trend of sharing STIX data has been rising since 2016.

We apply linear regression to our dataset to see the trend in the number of shared data for each source over time. We found that the volume of shared data for `AlienVault OTX`, `IBM X-Force Exchange`, `JamesBrine`, and `Cyware` gradually increased with $p < 0.05$. On the contrary, the volume of shared data for `Hail a TAXII` suddenly decreased with $p < 0.05$. However, this is because `Hail a TAXII` stopped sharing STIX data in June 2022. Furthermore, we observe that the total amount of shared data in STIX 2 gradually increases with $p < 0.05$, while the trend in the total amount in STIX 1 is not statistically significant.

Takeaway: STIX 1 is shared more often than STIX 2, but the sharing of STIX 2 data has been significantly increasing recently, along with new sources. The data is primarily sourced from leading security companies like `AlienVault OTX`, `Hail a TAXII`, and `IBM X-Force Exchange`, mainly through TAXII. This concentration implies that the open STIX dataset predominantly depends on a few sources. This concentration implies that the open STIX dataset largely relies on a few sources. While there is a low level of duplicate data across providers, substantial duplication is observed within individual providers. This emphasizes the need to remove duplicate data before deployment. Overall, we observe that the trend of sharing STIX data has been rising since 2016.

V. Timeliness

This section investigates the *timeliness* of STIX data, utilizing CTI-Lense. First, we analyze the relationship between the STIX data and corresponding security incidents. Then, we analyze the latency between the initial appearance of STIX data and its subsequent detection by popular scanning services.

A. STIX and Security Incidents

We examine the relevance of STIX data to actual security incidents. Timely threat information is crucial for consumers. Therefore, we first explore the latency between the date that security incidents are initially detected and the date when the corresponding STIX object is generated. Subsequently, we analyze the Granger causality [62] between the daily number of shared STIX objects and the daily number of security incidents reported from three distinct sources: Malpedia [16], security news websites, and the publication of CVEs. We perform this analysis at various time lags. Based on the time each security incident is detected, we measure the number of days until the corresponding information is first shared in the STIX data for each data source.

Security incidents. To measure the timeliness of STIX data sharing against security incidents, we focus on two infamous cyber attacks: the Mirai botnet [4] and the WannaCry ransomware [61]. Specifically, we measure the time (*i.e.*, the number of days) it took for the hash value of a malware sample for each attack to be shared as an attribute value in the STIX object, starting from the date of detection. We then compared

TABLE III: Number of days it takes for STIX data to be produced after a security incident occurs.

STIX sources	Mirai botnet		WannaCry	
	Initial	Variant	Initial	Variant
OTX AlienVault	55	10	96	30
IBM X-Force Exchange	43	30	109	50

the STIX data creation date delays of initial malware samples with their variants.

To identify initial malware sample hash values, we referred to the security reports of renowned companies for Mirai botnet (`6b7b6ee71c8338c030997d902a2fa593`) and WannaCry (`9c7c7149387a1c79679a87dd1ba755bc`) [39], [60]. To calculate the time delay, we searched the malware hash value on VirusTotal and determined its first detected time using *first_submission_time*. For malware variants, we randomly selected 100 malware hashes labeled Mirai and WannaCry in STIX data for each source and measured the average time difference with VirusTotal in the same way.

Table III summarizes our findings on the time delays. Only two sources, `AlienVault OTX` and `IBM X-Force Exchange`, contained initial malware sample hash information for Mirai and WannaCry. `IBM X-Force Exchange` produced the initial Mirai malware hash 43 days after the initial detection, which is 12 days faster than `AlienVault OTX`. In contrast, `AlienVault OTX` produced the initial WannaCry sample hash 96 days after the initial detection, which is 13 days faster than `IBM X-Force Exchange`. Unlike initial malware samples, we found that STIX data for the variants was shared more quickly for both the Mirai botnet and WannaCry. `AlienVault OTX` shared STIX data for Mirai botnet and WannaCry variants within an average of 10 and 30 days, respectively, while `IBM X-Force Exchange` shared it within 30 and 50 days, respectively.

Causality test. Analyzing the relationship between STIX objects and security incidents can be challenging due to missing attributes or unstructured text in the shared data. Alternatively, we collect daily security incidents from various sources and conduct causality tests with daily shared STIX data.

To obtain the daily number of security incidents, we collect 12,133 posts from Malpedia, 15,671 posts from popular news websites, and 151,431 published CVEs within the same period as the STIX dataset. Malpedia [16] is a resource for malware information, providing details such as malware family and threat actor information. Furthermore, it promptly shares information about security incidents from various sources, including well-known cybersecurity companies (*e.g.*, Avast, Bitdefender), news websites (*e.g.*, Threatpost), and blogs. This information includes reports of cyber attack campaigns such as targeted attacks and critical vulnerabilities.

We utilize Granger causality [62], a statistical concept of causality used to identify if one time series can predict another, to measure the relationship between STIX data and security incidents. We define a *time-lag* as the number of days between the date a security incident occurs and the date when the exchange rate of STIX data increases. Then, we analyze causality by varying the time lag from one day to 30 days. To address the multiple comparison problems, we apply Bonferroni corrections [45], with $n = 90$, since we analyze causality with a *time-lag* up to 30 days for three incident data sources. We find weak evidence of a causal relationship between security incidents reported on Malpedia and security news websites and the sharing of STIX data in our collected dataset. Specifically, we find that STIX data are shared after 2–4 days and 2–12 days, with $p < 0.05$, following the reporting of incidents in Malpedia and security news websites, respectively. However, we do not find any significant evidence suggesting a causal relationship between the publication of CVEs and STIX data sharing, with $p > 0.05$ for all *time-lag* values.

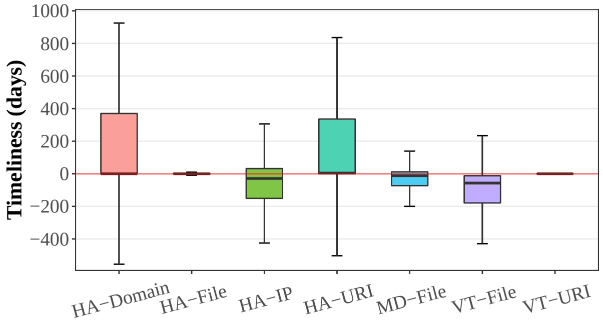


Fig. 5: Differences in the data generation time between STIX and each scanning service. VirusTotal (VT), HybridAnalysis (HA), and MetaDefender (MD).

B. STIX and Other Services

We also compare the generation time of STIX data with the submission time of corresponding data on popular scanning services, such as VirusTotal [71], HybridAnalysis [12], and MetaDefender [46]. These platforms serve as widely used repositories for virus samples. We concentrate specifically on the *Observable* attributes present in *Indicator* objects, as the *Observable* attributes contain one of four types of data: an IP address, a domain, a file hash, or a URL.

In total, we collect 70,997 file hashes and 374,973 URLs from VirusTotal, 31,290 file hashes from MetaDefender, and 14,924 domains, 10,179 file hashes, 836 IPs, and 8,055 URLs from HybridAnalysis. Then, we compare the *timestamp* attributes of the *Indicator* objects with the *first_submission_time* values of VirusTotal and *scan_start_time* values in HybridAnalysis and MetaDefender. We exclude the *Observable* types (*e.g.*, those in IP and domain objects for VirusTotal) for which the *first_submission_time* and *scan_start_time* are not shown in the results for each service.

Figure 5 shows the time differences between STIX data generation and scanning service data submission for each data type. For example, `HA-Domain` represents the number of days between when a domain data appeared in HybridAnalysis and the same domain data appeared in STIX. These differences are calculated by subtracting the STIX data generation time from the data submission time for each service. A concentration of positive values for timeliness indicates that STIX data generation is quicker than the submission time of the services (*e.g.*, `HA-Domain`). As shown in Figure 5, STIX data for domains, file hashes, and URIs is generated equally or more quickly than HybridAnalysis, while IP data is generated more slowly. STIX also shows comparable performance to VirusTotal for URI data on average, sharing 72% of URLs either before or on the same day. However, STIX is much slower than MetaDefender and VirusTotal with respect to the first submission date for file hash data. This is a critical issue, as antivirus response times can be slow, as reported in a previous study [8] that found an average antivirus response time of 19 days. These results suggest that utilizing STIX data could enhance the efficiency of prompt detection and defense against attacks involving domains and URIs. However, the efficacy of IP and file hash data is questionable.

Takeaway: Our analysis shows that producers of STIX data actively generate and share them within 2–4 days after a security incident occurs. However, except for domain and URI data, the efficacy of STIX is not yet confirmed, as most STIX data are generated considerably slower than other commercial services.

VI. Coverage

This section presents the evaluation of the application of the STIX standard in terms of *coverage*, utilizing CTI-Lense. Although STIX 1 and STIX 2 define 8 and 20 types of objects, respectively, only 6 (75%) and 15 (75%) object types are used in their respective datasets. This indicates that the full range of STIX objects is not being utilized. Examples of unused object types include *Campaign* and *Report* in STIX 1, and *Malware Analysis* in STIX 2.

TABLE IV: Objects and attributes used in the STIX dataset. We find that only 6 (75%) and 15 (75%) types of objects are used in the STIX 1 and STIX 2 datasets, which means that the full range of STIX objects is not being utilized.

STIX version	**Objects**		**Attributes**	
STIX 1	**Count**	**Prop.**	**Usage**	**Prop.**
Course of action	3,768	0.10%	5 / 19	26.32%
Exploit target	10,789	0.28%	6 / 15	40.00%
Incident	29,086	0.75%	4 / 34	11.76%
Indicator	3,846,499	98.77%	13 / 25	52.00%
Threat actor	719	0.02%	3 / 20	15.00%
TTP	3,734	0.10%	10 / 18	55.56%
STIX 2	**Count**	**Prop.**	**Usage**	**Prop.**
Attack pattern	1,662	0.07%	12 / 18	66.67%
Campaign	151	0.01%	9 / 20	45.00%
Course of action	1,181	0.05%	11 / 16	68.75%
Identity	1,110	0.04%	12 / 20	60.00%
Indicator	2,342,261	94.93%	18 / 23	78.26%
Intrusion set	211	0.01%	12 / 23	52.17%
Location	1	0.11%	9 / 25	36.00%
Malware	2,733	0.21%	16 / 26	61.54%
Observed data	5,087	0.00%	11 / 19	57.89%
Report	57,165	2.32%	14 / 19	73.68%
Threat actor	723	0.03%	10 / 27	37.04%
Tool	491	0.02%	11 / 20	55.00%
Vulnerability	5,755	0.23%	11 / 16	68.75%
Sighting	1,273	0.05%	12 / 22	54.55%
Relationship	47,666	1.93%	11 / 20	55.00%

We also analyze the dataset to identify the types of objects most frequently used by those producers (see Table IV). Based on our quantitative analysis, we obtained three important findings. First, we find that the most widely used object type is *Indicator*, which accounts for more than 90%, regardless of the versions of the STIX dataset. The reason why objects of *Indicator* type are so widely used is that they contain observables, such as hash values of malware or URL strings, which are used to identify specific malicious samples (see Figure 2). Second, we find some cases in which the number of similar data increases fast. For example, a *Cerber* ransomware variant was generated every 15 seconds [15] as the URLs used are frequently changed by domain generation algorithms (DGA). Finally, most STIX objects contain less than 50% of the attributes defined in STIX 1, while most STIX 2 objects use more than 50% of the attributes. Although it seems that the number of attributes used in STIX 2 is higher compared with that of STIX 1, we find that the main reason for such difference is that STIX 2 defines several additional but simple metadata, such as *Created*, *Modified*, and *Spec_version* as shown in Table V. We observe that an average of 2 and 6 attributes for STIX 1 and STIX 2 contain metadata, respectively. Other than that, the numbers of attributes used in STIX 1 and STIX 2 are similar, which means that producers still do not benefit from newly added objects and attributes.

Attributes usage in *Indicator*. As the *Indicator* type is the most used in the STIX dataset, we take a further look into the attributes used in *Indicator*. Figure 2 includes an example of an *Indicator* object in the STIX 1 dataset. *Title* and *Description* contain suspicious IP addresses and texts. *Type* and *Indicated_TTP* are used to describe the types of information (*e.g.*, malware family) with several pre-defined keywords, such as `IP Watchlist` (*i.e.*, suspicious IP addresses) and `ET.Evil` (*i.e.*, a name of a malware family). *Test_Mechanism* presents rules for detecting *Observable*.

Table V lists attributes with data on their usage in *Indicator*. Almost all the STIX 1 *Indicator* objects contain the *Title* (100%) and *Observable* (99.92%) attributes. However, not many objects include information types, such as *Type* (53.73%) or *Indicated_TTP* (34.76%). Furthermore, only 0.09% of *Indicator* objects contain some rules in the *Test_Mechanisms* (*e.g.*, Snort rule) attribute for detecting suspicious data. Similarly, most of the STIX 2 data includes the *name* (100%) and *pattern* (100%) attributes, which correspond to the *Title* and *Observable* attributes in STIX 1. By contrast, *labels* (32.62%) and *indicator_types* labels (17.13%) were seldom used or not used at all.

STIX object/attribute coverage over time. We analyze object and attribute usage for each source to analyze the coverage trend over time. STIX object type usage patterns across various sources show a degree of consistency, with occasional deviations. Figure 6 shows the monthly average number of objects used from each source from January 2015 to December 2022. For better visualization, we have excluded data from STIX sources with fewer than 100,000 unique objects. We observe that the number of object types and attributes provided by each source also increases overall, but there remains a limited number of object types over time.

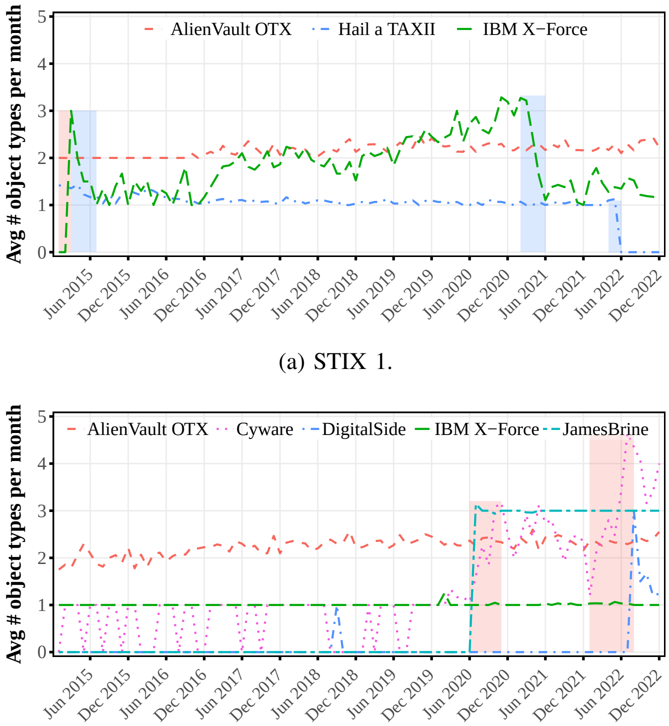
TABLE V: Attributes used in *Indicator* objects. We note that producers use only certain attributes to describe indicators, and few STIX data include security rules or information about attack steps useful to detect cyber attacks.

STIX version	Attributes	Count	Prop.
STIX 1	Id	3,846,499	100.00%
	Title	3,846,499	100.00%
	Observable	3,843,391	99.92%
	Timestamp	3,773,080	98.09%
	Description	2,346,868	61.01%
	Type	2,066,834	53.73%
	Producer	1,972,657	51.28%
	Version	1,899,238	49.38%
	Indicated_TTP	1,337,050	34.76%
	Confidence	1,308,416	34.02%
	Short_description	1,140,838	29.66%
	Test_Mechanism	3,482	0.09%
	Sightings	116	0.00%
STIX 2	Id	2,342,261	100.00%
	Created	2,342,261	100.00%
	Modified	2,342,261	100.00%
	Pattern	2,342,261	100.00%
	Type	2,342,261	100.00%
	Valid_from	2,342,261	100.00%
	Name	2,342,217	100.00%
	Description	2,170,475	92.67%
	Pattern_type	2,005,272	85.61%
	Spec_version	2,001,972	85.47%
	Pattern_version	1,999,546	85.37%
	Labels	763,935	32.62%
	Indicator_types	401,322	17.13%
	Kill_chain_phases	401,286	17.13%
	Created_by_ref	238,844	10.20%
	Object_marking_refs	98,700	4.21%
	Revoked	62,759	2.68%
	Valid_until	33,503	1.43%

For STIX 1, AlienVault OTX consistently used either *Indicator* and *Threat actor* or *Indicator* and *Incident*. IBM X-Force Exchange used up to three object types by December 2020, but since June 2021, it has primarily used only the *Indicator* type. Hail a TAXII used only the *Indicator* type since 2015 and until June 2022, when it appears to have ceased its sharing service.

For STIX 2, AlienVault OTX mostly shares two object types, with a slight increase in the average number recently. IBM X-Force Exchange consistently shares just one object type, while Hail a TAXII does not share any STIX 2 data. Cyware, JamesBrine, and DigitalSide showed a sudden increase in the number of objects used, indicating service initiation at that point. After June 2020, Cyware observed an increase in the average number of shared objects from 2–3 to about 4.

Our linear regression analysis indicated a statistically significant increase in the average number of STIX object types used across all sources, except for Hail a TAXII in STIX 1 and JamesBrine in STIX 2, with $p < 0.05$. Hail a TAXII showed a declining trend, likely due to the cessation of their service after June 2022.



(a) STIX 1.

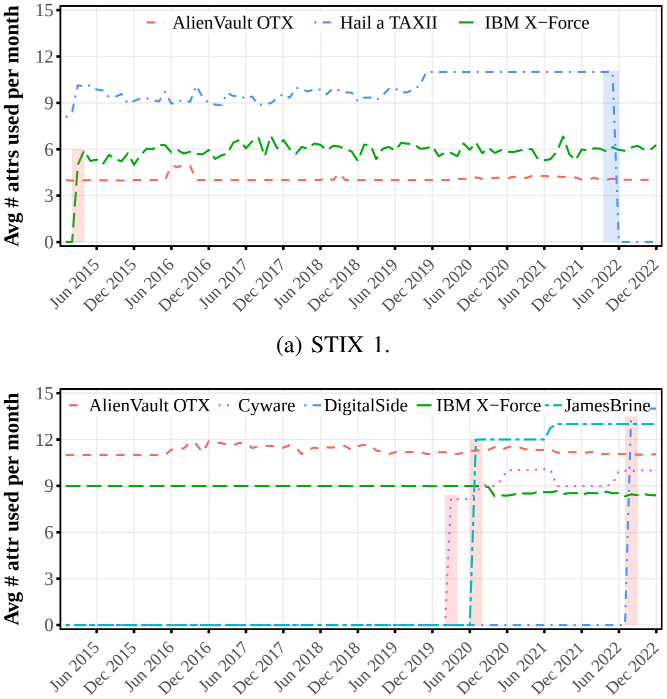
(b) STIX 2.

Fig. 6: Average number of STIX object types used for each source per month. The red zones indicate periods of rapid increase in STIX object types used, while the blue zone indicates a period of rapid decrease in STIX object types used.

STIX attribute usage patterns in *Indicator* objects show similar trends. Figure 7 shows the average number of attributes used in *Indicator* objects for each source per month. We performed linear regression on STIX attribute usage in *Indicator* objects per source and found a statistically significant increase in the average number of STIX attributes used across all sources ($p < 0.05$), except for AilenVault OTX in both STIX versions and IBM X-Force Exchange in STIX 2. AlienVault OTX showed no significant trend, and IBM X-Force Exchange gradually decreased its attribute usage ($p < 0.05$).

Interestingly, different sources use different object types and attributes to represent even the same CTI data, likely due to variations in STIX generation. Each source has its own unique web interface for collecting CTI information, with varying options and defaults. These differences appear to have led to unique object types and attribute patterns for each source.

Takeaway: Our analysis shows that the STIX data are mostly based on the *Indicator* objects, accounting for more than 90% of all the data. Most of the *Indicator* objects contain simple indicators of compromise information, such as malicious file hash or URL strings, while few of them include security rules to detect cyber attacks or information about attack steps to detect or predict multistep attacks. Also, we find that the number of object types and attributes used for STIX data is diverse depending on the data source.



(a) STIX 1.

(b) STIX 2.

Fig. 7: Average number of attributes used in *Indicator* objects for each source per month. The red zones indicate periods of rapid increase in attribute usage in *Indicator* objects, while the blue zone indicates a period of rapid decrease in attribute usage.

VII. Quality

This section analyzes the quality of STIX data by evaluating improper values and improper usage of objects/attributes, utilizing CTI-Lense.

A. Improper Value

We analyze whether the values in STIX objects and attributes written by producers are correct. We focus specifically on values in three objects for each STIX version – STIX 1: *Indicator*, *TTP*, and *Threat actor*; and STIX 2: *Indicator*, *Malware*, and *Threat actor*.

Figure 8 shows an example of an improper value case, where `Lazarus`, a threat group name, is described with the *Name* attribute of the *TTP* object. It is also important to ensure the *simple_hash_value* in the *Observable* attribute contains a malicious file hash value related to `Lazarus`. For this example, we examine whether the hash value `12cc14bbbc421275c3c6145bfa186dff` is actually related to *TTP*.

In addition, the *simple_hash_value* in the *Observable* attribute must contain a malicious file hash value, which is related to `Lazarus`, to use STIX properly. Incorrect STIX data can cause severe problems, leading to improper responses to security threats. Rectifying incorrect STIX data can be a heavy burden for security practitioners who process large volumes of threat data.

```
<stix:Indicator id="indicator-...">
  <indicator:Observable idref="...">
    <cyboxCommon:simple_hash_value>
      12cc14bbbc421275c3c6145bfa186dff
    </cyboxCommon:simple_hash_value>
  </indicator:Observable>
  <indicator:Indicated_TTP>
    <stix:TTP id="ttp-...">
      <ttp:Behavior>
        <ttp:Malware_Instance>
          <ttp:Name> Lazarus </ttp:Name>
        </ttp:Malware_Instance>
      <ttp:Behavior>
      ...
    </stix:TTP>
  </indicator:Indicated_TTP>
</stix:Indicator>
```

Fig. 8: Example of improper value of an attribute. Note that Lazarus, a threat actor, is assigned to the *Malware_Instance* attribute in the *TTP* object.

Incorrect keywords in attributes. To evaluate the improper values of attributes, we first focus on the mapping between attributes and values in three objects – *Threat actor*, *TTP*, and *Malware* – where values are a form of attribute keywords. To quantify the level of improper values, we first generate the ground truth dataset of threat actors and malware families by querying three data sources: Malpedia [16], MITRE ATT&CK [47], and MISP GitHub[4], which are widely used for labeling threat actors or malware families [67], [59], [32]. In detail, for each STIX data in the datasets, we extract the values of the *Name* attributes of the *Threat actor*, *TTP*, and *Malware* and make a list of names. As there can be different keywords that indicate the same target (*e.g.*, "Lazarus" and "Hidden Cobra"), we also refer to the "alias" fields and include all the values of the field in our keyword list. Finally, the list includes 2,111 threat actor names and 4,071 malware family names. Then, we search all the names in the list of the data sources and classify the results into one of *Threat actor*, *TTP*, *Malware*, or *Unknown*. Also, we collect naming conventions of threat actors (*e.g.*, "UNC N", "APT-C-N," and "Magecart group N") used by security vendors. Based on the ground truth dataset and naming conventions, we determine the mapping between attributes and values in STIX data is correct if the mapping matches any entry in the ground truth dataset or a value in the *Name* attribute of the *Threat actor* follows any of the naming conventions. For instance, if we find "UNC 10" in the *Name* attribute of the *Threat actor*, we consider it correct as the name follows the naming convention "UNC N."

We report our analysis results in Figure 9, where **Correct** means that the values written in the STIX data and in the ground truth dataset are in the same objects, or the values follow any of the naming conventions, while **Incorrect** means that the values of the STIX data and that of the ground truth dataset are different. We mark as **Unmatched** the values classified as *Unknown* in our ground truth dataset. We find that 19% of *Threat actor* objects of both STIX versions contain *TTP* names or *Malware* names. Furthermore, 62% and 58% of malware family information in STIX 1 and 2 (*i.e.*, *TTP* and *Malware*) does not match any of the entries

[4] https://github.com/MISP/misp-galaxy/tree/main/clusters

TABLE VI: Detection rates of *Indicator* objects in the STIX dataset by observable type and commercial scanning service. **Detected** denotes the proportion of *Indicator* objects detected by at least one engine out of the total count of requests. **Not det.** denotes the proportion of *Indicator* objects detected by no engine out of the total count of requests. **N/A** denotes the proportion of *Indicator* objects that do not exist in a scanning service.

		VirusTotal			**HybridAnalysis**			**MetaDefender**		
Observable types	**Count**	**Detected**	**Not det.**	**N/A**	**Detected**	**Not det.**	**N/A**	**Detected**	**Not det.**	**N/A**
AddressObjectType	43,537	50.62%	49.38%	0.00%	8.75%	91.25%	0.00%	30.02%	69.98%	0.00%
DomainNameObjectType	163,121	39.99%	59.90%	0.12%	9.25%	90.63%	0.12%	18.28%	81.59%	0.13%
FileObjectType	88,470	78.37%	1.87%	19.75%	42.24%	5.30%	52.46%	77.02%	3.22%	19.75%
URIObjectType	377,857	97.06%	2.18%	0.76%	2.37%	96.87%	0.77%	91.72%	8.26%	0.02%
Total	672,985	77.77%	19.18%	3.05%	9.69%	82.95%	7.35%	67.99%	29.37%	2.64%

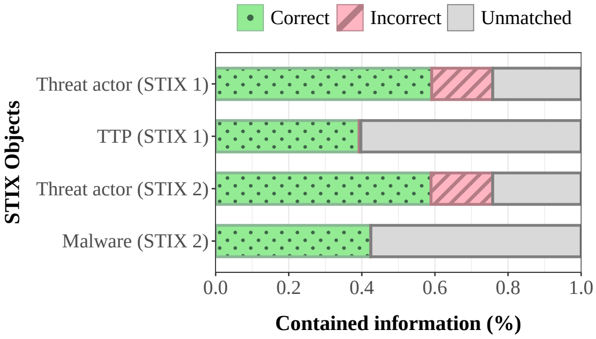


Fig. 9: Proportion of correct and incorrect attribute values. We find that 19% of *Threat actor* objects of both STIX versions contain *TTP* names or *Malware* names.

in the ground truth dataset. To understand the reason for such unmatched objects, we manually analyze their values. We find that the majority of the values are named based on producers' own conventions. For instance, some STIX producers use the substrings (*e.g.*, "DIK" and "ZBOT.DIK") of the name of malware (*e.g.*, "TSPY_ZBOT.DIK") defined by security providers (*e.g.*, TrendMicro). There are also some misspelled cases (*e.g.*, "WannaCry" as "Wancry") that increase the number of unmatched cases.

Validation of values in attributes. We investigate whether the values in *Observable* attributes (*e.g.*, malware hashes, domains, URLs, and IPs), generated between January 1, 2020, and January 3, 2022, are actually malicious.

To build a ground truth dataset of *Indicator* objects, we retrieve scanning reports from three services: VirusTotal, HybridAnalysis, and MetaDefender, which are widely used to validate threat indicators [28], [38], [37], [21], [55]. For each attribute type (*i.e.*, malware hashes, domains, URLs, and IP addresses), VirusTotal and MetaDefender provide detection results from various anomaly detection engines, and HybridAnalysis provides a threat score with three status tags (*i.e.*, malicious, suspicious, and no specific threat). We used the report, search, and lookup APIs provided by VirusTotal, HybridAnalysis, and MetaDefender, respectively, to obtain the latest scanning results. We collected the values for different observable types of indicators, including hashes of malware, domains, URLs, and IP addresses, which correspond to the values of the `FileObjectType`, `DomainNameObjectType`, `URIObjectType`, and `AddressObjectType` properties of *Indicator* objects, respectively. Since the STIX data we searched for may not exist for each service, we measure the ratio of detected, undetected, and not applicable (N/A) data for each service.

Table VI summarizes the statistics derived from reports collected from three services. In total, we query 672,985 *Observable* data in the *Indicator* objects and obtain 652,452 (96.95%), 623,492 (92.65%), and 655,213 (97.36%) reports for VirusTotal, HybridAnalysis, and MetaDefender, respectively. We first count the number of detections as the number of reports that include at least one positive result. Then, we compute the detection rate by dividing the detection count by the number of total reports. We find that 523,352 (77.77%) and 457,595 (67.99%) threats described by the corresponding STIX data are confirmed by at least one of the engines in VirusTotal and MetaDefender, respectively. However, only 65,225 (9.69%) STIX data are confirmed by HybridAnalysis. In detail, the `FileObjectType` data, which is applicable in each source, can be almost found in all three scanning services with high accuracy (97.67%, 88.85%, and 95.99% for VirusTotal, HybridAnalysis, and MetaDefender, respectively). In addition, the `URIObjectType` data, which is the most dominant observable type (56.15%) in STIX, can be almost found in VirusTotal (99.24%) and MetaDefender (99.98%) with high accuracy (97.80% and 91.72%, respectively). However, although most (over 99%) of the `AddressObjectType` data and `DomainNameObjectType` data exist in all three scanning services, their detection rates are low (lower than 50.62%). On the other hand, the `FileObjectType` data are the most missing type in VirusTotal (*i.e.*, up to 52.46% of them are missing), but they show the highest accuracy (up to 96.95%).

We analyze the VirusTotal detection result of 672,985 STIX data in more detail. To this end, for each STIX data, we check how many detection engines classify it as an anomaly. We evaluate accuracy based on the threshold [64], which indicates the minimum number of anomaly detection engines that report a positive. Since the detection results for individual VirusTotal engines can be flipped over time, we measure the accuracy of the data with respect to the various thresholds for each *Observable* attribute type. We select thresholds that range from $t = 2$ to 39, as suggested by [64].

Figure 10 shows that the `FileObjectType` achieves over 90% accuracy when the threshold is set to less than

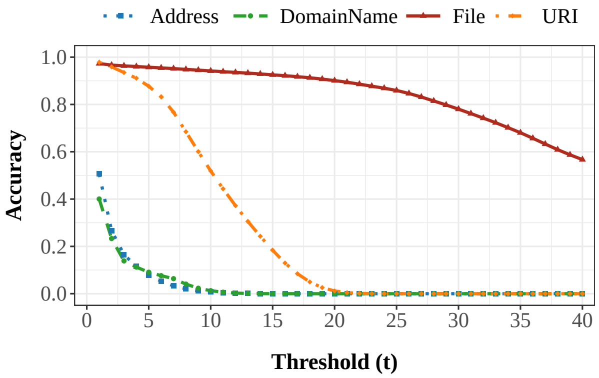


Fig. 10: Accuracy for observable types based on each threshold t. We note that both `FileObjectType` and `URIObjectType`, which account for 70.34% of all the data, achieve about 90% of accuracy when the threshold is set to less than or equal to $t = 5$ (96.74% and 87.69%, respectively). Although STIX does not always contain accurate values, STIX includes quite verified values for file objects and URIs.

or equal to $t = 20$ (90.08% for $t = 20$). We conclude that STIX disseminates accurate values, especially for file objects. Subsequently, `URIObjectType` achieves about 90% accuracy in a small range with a threshold less than or equal to $t = 5$ (87.69% for $t = 5$). However, the detection rate becomes 0, when the threshold is set to over $t = 20$. In addition, we find that about 50% of `AddressObjectType` and `DomainObjectType` data are detected by no engine in VirusTotal. It means that for the `AddressObjectType` and `DomainObjectType` data, STIX data are not confirmed to be accurate with regard to VirusTotal; however, it does not mean that the STIX data are inaccurate because there can be some addresses or domain names that the VirusTotal engines do not detect. Interestingly, when $t = 20$, the accuracy of most observable types becomes 0, but `FileObjectType` shows significantly high accuracy. The result indicates that STIX provides verified information on file objects.

Validation of values in attributes by producer. We were curious to see if the validity of STIX data varies depending on the producer. To this end, we divided the *Indicator* objects by producer and compared them with VirusTotal detection results. Table VII compares the validity of STIX *Indicator* objects shared on VirusTotal from January 2020 to January 2022 by producer. We categorized the *Indicator* objects into three groups: those from producers with known identities (`phishtank.com` and `dshield.org`) and those without producer information ('None').

Our results are based on VirusTotal scanning reports of 672,985 *Indicator* objects shared over two years. Of the 328,600 (48.83% out of 672,985) *Indicator* objects with producer names, 328,585 objects (99.99%) are produced by `phishtank.com` and achieve a very high accuracy of 99.79%, which is much higher than *URIObjectType* objects without producer names (77.74% out of 49,963)[5]. This suggests that the accuracy of *URIObjectType* objects with producer names is higher than that of objects without producer names.

TABLE VII: Validity of *Indicator* objects in the STIX dataset, grouped by producer identity, using VirusTotal. *Detected* denotes the proportion of *Indicator* objects detected by at least one engine in the VirusTotal.

Producer	Observable types	# (%) Objects	Detected
phishtank.com	URIObjectType	328,585 (99.99%)	99.79%
	AddressObjectType	-	-
dshield.org	URIObjectType	-	-
	AddressObjectType	15 (0.00%)	6.67%
None	URIObjectType	49,963 (14.51%)	77.74%
	AddressObjectType	43,522 (12.64%)	50.63%
	Total	344,385 (100%)	-

TABLE VIII: Correctly mapped STIX objects with APT reports.

Indicator attr.	Ref. object	Overlap	# (%) Correct
Observable (STIX 1)	Threat actor	17,737	15,875 (89.50%)
	TTP	10,632	3,601 (33.87%)
Pattern (STIX 2)	Threat actor	27,352	12,538 (45.84%)
	Malware	1,661	86 (5.18%)

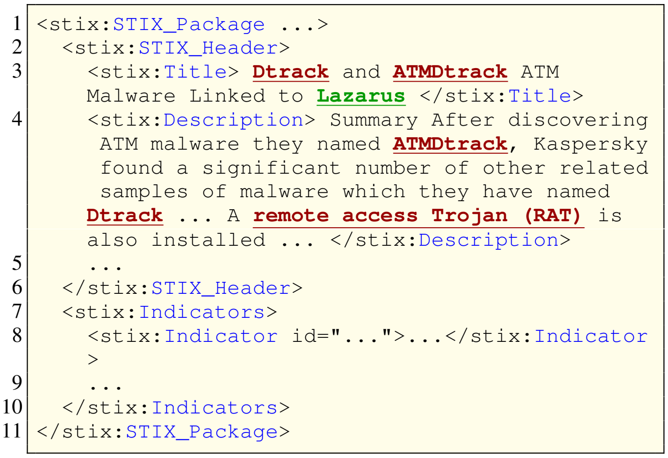
However, in the case of *AddressObjectType* objects, the situation is reversed. The 15 objects on `dshield.org` have a very low accuracy of 6.67%, which is significantly lower than objects without producer names (50.63%). We believe that the *AddressObjectType* objects may generally have incorrect (or no longer valid) information, as the objects are used to specify IP addresses, and IP addresses are volatile and temporarily used for malicious purposes. Also, this case is difficult to generalize, as the sample size is only 15.

Correctness of attributes mapping. Finally, we validate the values of *Observable* attributes such as *Threat actor*, *TTP*, and *Malware*. For example, in Figure 8, we examine whether the value of *TTP* (`12cc14bbbc421275c3c6145bfa186dff`) is truly linked to Lazarus.

To conduct this analysis, it is necessary to verify the validity of each *Observable* attribute value. We use threat reports written by security experts for the ground truth. In other words, we verify the validity of an *Observable* (STIX 1) or a *Pattern* (STIX 2) attribute based on the analysis results presented in these threat reports. We collect 4,042 threat reports from renowned security companies such as Kaspersky and TrendMicro. These reports contain a total of 57,382 *Observable* (STIX 1) or *Pattern* (STIX 2) attribute values within *Indicator* objects that overlap with our dataset. Table VIII shows the number and proportion of valid attribute values verified by the threat reports. Interestingly, approximately 90% of *Threat actor* attributes in STIX 1 are valid. However, for `TTP` attributes in STIX 1, only about 34% have valid values. Furthermore, the rate of valid attribute values for `Threat actor` and `Malware` attributes in STIX 2 is relatively lower, standing at 45.84% and 5.18%, respectively.

The results of our analysis highlight a substantial mismatch between the attributes of STIX data, excluding *Threat Actor*,

[5] Note that a total number of objects without producers is 344,385 (51.17% out of 672,985), but only 49,963 (14.51%) has *URIObjectType* objects.



```
<stix:STIX_Package ...>
  <stix:STIX_Header>
    <stix:Title> Dtrack and ATMDtrack ATM
    Malware Linked to Lazarus </stix:Title>
    <stix:Description> Summary After discovering
     ATM malware they named ATMDtrack, Kaspersky
     found a significant number of other related
     samples of malware which they have named
    Dtrack ... A remote access Trojan (RAT) is
    also installed ... </stix:Description>
    ...
  </stix:STIX_Header>
  <stix:Indicators>
    <stix:Indicator id="...">...</stix:Indicator
    >
    ...
  </stix:Indicators>
</stix:STIX_Package>
```

Fig. 11: Example of *improper usage* of an attribute. ATMDtrack, the name of malware, is described in the *Description* attribute rather than the *Malware_Instance* attribute.

in STIX 1 and the IoCs reported in threat reports. The discrepancy between STIX data and real-world threat reports suggests that we need better data quality control mechanisms. This includes standardized procedures, data validation, and ongoing monitoring. We may also need improved methodologies for generating STIX attributes.

B. *Improper Usage*

We measure the number of objects that contain information, which can be precisely represented with other objects or attributes, within a sentence, but does not include the objects or attributes. For instance, Figure 11 presents an example of a STIX header containing a narrative description of a malicious sample, which we defined as improper usage. Specific actions such as Dtrack, ATMDtrack, and RAT are described in *Title* and *Description* but not in a *TTP* object that is defined to describe such actions. Similarly, the threat actor Lazarus should be represented using the *Threat actor* object, but it is only described in *Title* of *STIX_Header*. We also find this practice in the STIX 2 dataset. Many STIX 2 data (*e.g.*, threat actor and malware) use the *description* attribute in *Report* objects to describe specific information in a narrative way. Although such attribute usage in *Title* or *Description* does not violate the standard, this practice limits the main purpose of automatic processing and rapid response to threats by machines as intended by STIX. To this end, we first make a list of keywords from *Attack pattern*, *Malware instance*, *Target information*, and *Threat actor*. We next review sentences of the *Description* attribute of the *Indicator* objects and collect the objects that include any keyword in the list. From the objects, we classify an object in which the keyword is not described with the precise objects or attributes as improper usage.

We make a list of keywords based on the list of keywords used in the improper usage and the vocabularies, which are pre-defined keywords for specific attributes listed in the STIX standard[6]. We also add all the words used in the **Attack pattern** objects (*e.g.*, Brute Force) and the **Target information** objects (*e.g.*, User information or Google Inc.) as their values are simple keywords. With the list, we check the *Indicator* objects if they are improperly used.

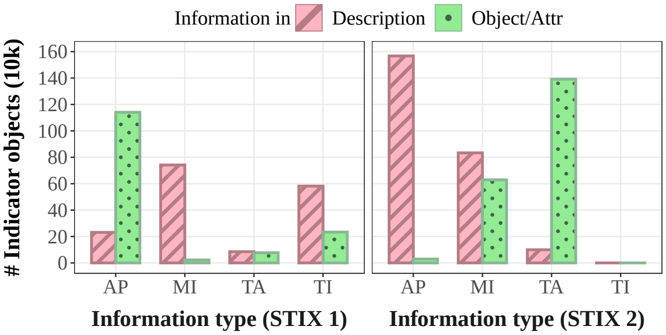


Fig. 12: Number of indicator objects where information is written in the *Description* attribute and the precise object/attribute for four information types: *Attack pattern* (AP), *Malware instance* (MI), *Threat actor* (TA), and *Target information* (TI). All the types of the *Indicator* objects but *Attack pattern* imprecisely use attributes in STIX 1 and 98% of the *Indicator* objects in STIX 2 include information about attack patterns only in the *Description* attribute.

Our analysis shows that all the types of the *Indicator* objects but *Attack pattern* improperly use attributes in STIX 1 (see Figure 12). Only less than 45% of *Indicator* objects, which include keywords in the description sentence, precisely represent the information with objects or attributes for all types. Furthermore, 98% of the *Indicator* objects in STIX 2 describe attack pattern information only in a narrative way. The result shows that producers rely on description rather than strictly using the specified attributes in describing the threat. Notice that this practice undermines machine readability, which is one of the main purposes of STIX.

Takeaway: Our analysis shows that producers do not properly use objects or attributes as they are intended in the STIX standard. We report two common misusages. First, STIX does not always contain accurate values, although STIX includes quite verified values for file objects and URIs. Second, producers often do not properly select objects and attributes to represent values. Instead, they describe specific information in a narrative way, rather than using specific STIX attributes.

VIII. Discussion

In this section, we discuss the impact of our findings and recommendations to move forward.

A. *Impact of Findings*

Our findings have several important implications for the use of STIX data in cybersecurity.

Volume. The low volume of STIX data generated and shared daily (an average of only 2,063 unique STIX objects) suggests that security service providers have not yet widely adopted STIX. This is a significant limitation, as cybersecurity professionals do not have enough STIX data to meet their

[6] https://stix.mitre.org/XMLSchema/default_vocabularies/1.2.0/stix_default_vocabularies.xsd

needs. Additionally, the high duplication rate among STIX data objects suggests that it is essential to remove duplicate STIX data.

Timeliness. The prompt sharing of URL objects is a positive finding, suggesting that STIX can be effectively used to detect new attack websites. However, the slower sharing of file-type threat objects is a concern, suggesting that STIX is not yet used effectively to prevent malware attacks.

Coverage. The limited number of STIX data types used in practice suggests that the STIX standard is underutilized, potentially due to its complexity. This is a missed opportunity, as STIX can represent a wide range of sophisticated cybersecurity threats. The lack of security rules in STIX indicators is also a significant limitation, as it hinders the automation of cyber attack detection.

Quality. Incorrect threat actor attribution and the frequent use of narrative text in attack pattern objects can lead to ineffective countermeasures [69] and make automatic processing difficult. This requires security analysts to manually verify the validity of CTI data, which is a time-consuming task that requires security knowledge and experience. For example, a previous study [50] showed that three full-time annotators spent five months labeling 133 reports. Such extensive manual efforts can delay timely responses to cyber threats [52].

B. Recommendations

Our findings suggest several actions organizations can take to promote the adoption of STIX:

Common vocabulary. The STIX standard includes several predefined vocabularies to describe information related to STIX objects. For example, the TTP object includes a type attribute that represents malware types, using 18 keywords (*e.g.*, `Adware`, `Bot`, and `Ransomware`). However, our analysis revealed that only two keywords (`Bot Loader` and `Remote Access Trojan`) were observed in the collected TTP objects. Instead of using existing keywords, many producers use their own words to describe threat information, as shown in previous work [58], [59]. This leads to confusion among consumers and makes security rule-generation tasks especially difficult to automate. To address this problem, the security community should conduct further analysis to represent and organize such information effectively. Additionally, we must bring together a consortium of organizations using STIX to build a universal and comprehensive standardized vocabulary for CTI. Specifically, similar to how new vulnerabilities receive new identifiers in databases such as CVE, a public database should also be developed for malware.

Training program. Recall that there are various objects and attributes specified in the STIX standard, but only 75% (STIX 1) and 75% (STIX 2) of objects and less than 70% of attributes are used for most objects. Furthermore, our analysis shows that producers often improperly use objects and attributes when describing threats, indicating that producers and consumers still need to fully benefit from the proper expressiveness of STIX. Organizations for standardization (*e.g.*, OASIS) and service providers need to consider developing educational programs to train security analysts who use STIX data. They are recommended to offer practical guidelines for creating and sharing CTI data in the STIX and TAXII standards.

Automated tools. Our analysis shows that many STIX data contain common errors, such as spelling mistakes, imprecise object and attribute usage, and duplicate data. To detect and reduce these errors, service providers could deploy automated verification and deduplication processes. Alternatively, specialized software could be developed and distributed to help producers generate structured STIX data from raw sources. Such tools can help security analysts avoid human errors and inconsistencies in STIX objects and attributes during the data generation process.

Accountability. One way to enhance the trustworthiness of STIX data is by upholding the reputation of security analysts who produce this data. Currently, anyone can anonymously create and modify STIX data. Although the *Producer* attribute is specified in the *Indicator* object of the STIX standard, it is included in only 51.28% of STIX 1 data. The integrity of this attribute is not ensured, complicating the task of tracing the authors of the STIX data. As the results in Table VII demonstrate, STIX objects that include producer names tend to be more accurate. Standard organizations might consider promoting the use of the *Producer* attribute to motivate STIX producers to explicitly add their identities. Furthermore, introducing a new attribute representing the signature of the identities to ensure the integrity of the STIX producers' identities is essential. Maintaining and tracking historical records regarding the validity of shared STIX data can motivate security analysts to produce more reliable data.

C. Limitations

First, although we attempted to collect data from all the TAXII servers and public repositories listed in the STIX official website [43], we only found ten available STIX data sources. It may introduce some biases in our analysis, and more STIX data may help for generalization. We excluded commercial and industrial resources of STIX data because we set up our scope on publicly available STIX data. As future work, it would be interesting to conduct a study analyzing the differences and similarities between the public STIX and commercial and industrial datasets.

Second, we conducted a keyword-based analysis, which may not be accurate, in finding the imprecise usage of the keywords in narrative descriptions. To verify the validity of the keyword-based analysis, we randomly selected 100 narrative descriptions and evaluated the imprecise usage of keywords (see Figure 12). Then, we manually checked that the keywords were correctly used in the description, and our approach achieved 87% accuracy. However, more effective methods could be applied to properly extract related information from narrative descriptions, such as topic modeling in text analysis.

Third, CTI-Lense analyzes the generation and dissemination of STIX data. However, it does not track the adoption and utilization of the STIX data by various companies. The emphasis is primarily on determining the types and extent of STIX data shared across platforms. The subsequent consumption of these data remains unexamined for future work.

Lastly, the lack of ground truth is one of the most challenging issues in our analysis. To obtain more reliable and accurate actual security threat data, we use at least three representative services related to each measurement (*e.g.*, VirusTotal, HybridAnalysis, and MetaDefender for the timeliness analysis) as sources of security threat data instead of relying on a single service. Note that these sources are widely used as ground truth in prior works [28], [38], [37], [21], [55], [67], [59], [32].

IX. Related work

Quality and applicability assessment of CTIs. Existing research has not focused much on analyzing the quantity and quality of CTI data. Li et al. [26] conducted a quantitative evaluation of the coverage and accuracy of various IoC data, finding significant overlap in threat intelligence data from various public sources. Similarly, Griffioen et al. [27] assessed the timeliness, sensitivity, originality, and impact of 1.38 million indicators from threat intelligence feeds, highlighting varying response times, with some feeds providing indicators within days of a security incident and others lagging behind by months. Bouwman et al. [73] emphasized the practical effectiveness of publicly available CTI data, showing that blocklists from public CTI data sources were more successful at thwarting COVID-19 related phishing attacks than existing defenses. However, these studies focus on a limited range of threat intelligence types, such as simple IoC types (IPs and malware hashes) [26], [27] or a specific threat intelligence source [73]. Additionally, they do not consider any advanced threat data, such as TTPs and threat actors. Unlike these existing studies, we focused our research on how STIX objects, a type of structured CTI data, are being shared.

Automated extraction and generation of CTIs. Several attempts have been made to automate the extraction and generation of structured CTI data from unstructured data. For instance, Kim et al. [36] introduced CyTIME, a CTI management framework that integrates heterogeneous CTI data into a single, structured STIX format and auto-generates security rules. Sadique et al. [22] proposed a privacy-preserving mechanism that utilizes Syslog data and represents raw cyber threat data in STIX format in an automated manner. However, these frameworks generate structured CTI data that contains only simple IoC information. To address this limitation, several frameworks have been proposed to generate structured data, including advanced data such as threat actors and IoC categories, from unstructured data. For example, Kim et al. [14] proposed CTIMiner, which parses public security reports and generates structured CTI data for specific domains (*e.g.*, finance and IoT). Similarly, Koloveas et al. [49] proposed INTIME to identify and analyze cyber threats in the IoT domain. Zhao et al. [34] proposed machine learning-based methodologies for creating highly accurate CTI data in an automated manner, leveraging Convolutional Neural networks (CNNs), word embedding, and syntactic dependency. Fujii et al. [63] proposed a method to generate CTI data using several state-of-the-art natural language processing techniques, such as named entity recognition and relation extraction, to extract named entities and relations between them. Despite these efforts to generate sophisticated structured CTI data, the best-performing frameworks still achieve F1 scores of around 80%.

X. Conclusion

In this paper, we analyze STIX usage in terms of volume, timeliness, coverage, and quality. Although STIX is frequently shared during security incidents, there is still room for improvement in practice to fully benefit from STIX's diverse expressive power and the quality of values in STIX data. Based on our findings in the collected STIX dataset, we suggest four practical recommendations: (1) establishing a common vocabulary to ensure the consistency and comparability of STIX data across different entities; (2) developing training programs to encourage security analysts to generate more valid and useful STIX data; (3) using tools that automatically generate STIX data from raw data or natural language reports, and validate the objects and attributes in STIX data; and (4) maintaining and tracking the historical records on the validity of shared STIX data to encourage security analysts to generate more credible and trustworthy data.

Acknowledgment

The authors would thank anonymous reviewers and the shepherd. Hyoungshick Kim is the corresponding author. This work was supported by Korea Internet & Security Agency (KISA) grant (No.1781000003, Development of a Personal Information Protection Framework for Identifying and Blocking Trackers) and Institute for Information & communication Technology Planning & Evaluation (IITP) grant funded by the Korea government (NIST) (No.2018-0-00532, Development of High-Assurance (>=EAL6) Secure Microkernel, 40%), (No.2019-0-01343, Regional Strategic Industry Convergence Security Core Talent Training Business), and (No.2022-0-00495, On-Device Voice Phishing Call Detection). This work was also supported by National Science Foundation (2210137 and 2335798), Science Alliance's StART program and gifts from Google exploreCSR and TensorFlow, and the KENTECH Research Grant (202200048A).

References

[1] Abnormal, "Fraudsters use email in phone fraud scams, targeting 89% of organizations," 2021, access date: 28 July 2022. [Online]. Available: https://medcitynews.com/uploads/2022/03/Abnormal-Security_H2-2021_Email-Threat-Report.pdf

[2] Accenture, "The cost of cybercrime," 2019, access date: 28 July 2022. [Online]. Available: https://www.accenture.com/_acnmedia/PDF-96/Accenture-2019-Cost-of-Cybercrime-Study-Final.pdf

[3] R. Andrew, S. Stavros, and K. Nicholas, "A comparative analysis of cyber-threat intelligence sources, formats and languages," *Electronics*, vol. 9, p. 824, 2020.

[4] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis *et al.*, "Understanding the mirai botnet," in *Proceedings of the 26th USENIX Security Symposium*, 2017, pp. 1093–1110.

[5] S. Appala, N. Cam-Winget, D. McGrew, and J. Verma, "An actionable threat intelligence system using a publish-subscribe communications model," in *Proceedings of the 2nd ACM Workshop on Information Sharing and Collaborative Security*, 2015, pp. 61–70.

[6] AV-TEST, "Malware statistics," 2023, access date: 15 May 2023. [Online]. Available: https://www.av-test.org/en/statistics/malware/

[7] H. Booth, D. Rike, and G. A. Witte, "The national vulnerability database (NVD): Overview," 2013.

[8] M. Botacin, F. Ceschin, P. De Geus, and A. Grégio, "We need to talk about antiviruses: challenges & pitfalls of av evaluations," *Computers & Security*, vol. 95, p. 101859, 2020.

[9] J. Bret, P. Rich, and D. Trey, "OASIS Standard - STIX Version 2.1," 2022, access date: 15 December 2022. [Online]. Available: https://docs.oasis-open.org/cti/stix/v2.1/os/stix-v2.1-os.pdf

[10] R. Brown and R. M. Lee, "The evolution of cyber threat intelligence (CTI): 2019 sans cti survey," *SANS Institute*, 2019.

[11] M. Corporation, "Cyber observable eXpression (CybOX)," 2017, access date: 23 May 2022. [Online]. Available: http://cyboxproject.github.io/releases/2.1/

[12] CrowdStrike, "Hybrid Analysis," 2012, access date: 27 May 2022. [Online]. Available: https://www.hybrid-analysis.com/

[13] W. Cynthia, D. Alexandre, W. Gérard, and I. Andras, "Misp: The design and implementation of a collaborative threat intelligence sharing platform," in *Proceedings of the 3rd ACM on Workshop on Information Sharing and Collaborative Security*, 2016, pp. 49–56.

[14] K. Daegeon and K. H. Kang, "Automated dataset generation system for collaborative research of cyber threat analysis," *Security and Communication Networks*, vol. 2019, 2019.

[15] N. Daniel, "A behavioural-based approach to ransomware detection," *Whitepaper. MWR Labs Whitepaper*, 2017.

[16] P. Daniel, C. Martin, E. Steffen, and P. Elmar, "Malpedia: a collaborative effort to inventorize the malware landscape," *The Journal on Cybercrime & Digital Investigations*, 2017.

[17] S. Daniel, B. Fabian, C. Marco, and P. Günther, "Measuring and visualizing cyber threat intelligence quality," *International Journal of Information Security*, vol. 20, pp. 21–38, 2021.

[18] S. Dave, "Cyber threat intelligence uses, successes and failures: The SANS 2017 CTI survey," *SANS Institute*, 2017.

[19] de Melo e Silva Alessandra, C. G. J. José, de Oliveira Albuquerque Robson, and G. V. L. Javier, "A methodology to evaluate standards and platforms within cyber threat intelligence," *Future Internet*, vol. 12, p. 108, 2020.

[20] Deep-Instinct, "Cyber threat landscape report," 2020, access date: 28 July 2022. [Online]. Available: https://info.deepinstinct.com/en/tof/cyber-threat-report-2021?_ga=2.149360946.587810922.1658982278-1327454874.1658982278

[21] P. Dodia, M. AlSabah, O. Alrawi, and T. Wang, "Exposing the rat in the tunnel: Using traffic analysis for Tor-based malware detection," in *Proceedings of the 29th ACM SIGSAC Conference on Computer and Communications Security*, 2022, pp. 875–889.

[22] S. Farhan, C. Sui, V. Iman, B. Shahriar, and S. Shamik, "Automated structured threat information expression (STIX) document generation with privacy preservation," in *Proceedings of the 9th IEEE Annual Ubiquitous Computing, Electronics & Mobile Communication Conference*, 2018, pp. 847–853.

[23] P. Foremski and P. Vixie, "The modality of mortality in domain names," *Virus*, p. 1, 2018.

[24] S. Gallagher, "Locky: crypto-ransomware rides in on malicious word document macro," 2016, access date: 15 December 2022. [Online]. Available: https://arstechnica.com/information-technology/2016/02/locky-crypto-ransomware-rides-in-on-malicious-word-document-macro/

[25] Y. Ghazi, Z. Anwar, R. Mumtaz, S. Saleem, and A. Tahir, "A supervised machine learning based approach for automatically extracting high-level threat intelligence from unstructured sources," in *Proceedings of the 16th International Conference on Frontiers of Information Technology*. IEEE, 2018, pp. 129–134.

[26] L. V. Guo, D. Matthew, P. Paul, M. Damon, V. G. M, and S. Stefan, "Reading the tea leaves: A comparative analysis of threat intelligence," in *Proceedings of the 28th USENIX Security Symposium*, 2019, pp. 851–867.

[27] G. Harm, B. Tim, and D. Christian, "Quality evaluation of cyber threat intelligence feeds," in *Proceedings of the 18th International Conference on Applied Cryptography and Network Security*, 2020, pp. 277–296.

[28] D. Hitaj, G. Pagnotta, B. Hitaj, L. V. Mancini, and F. Perez-Cruz, "MaleficNet: Hiding malware into deep neural networks using spread-spectrum shannel coding," in *Proceedings of 27th European Symposium on Research in Computer Security*. Springer, 2022, pp. 425–444.

[29] J. Hurley, "Threat post," 2009, access date: 15 May 2023. [Online]. Available: https://threatpost.com/

[30] IBM, "What is threat hunting?" access date: 22 June 2023. [Online]. Available: https://www.ibm.com/topics/threat-hunting

[31] M. Jason, "Using network based security systems to search for STIX and TAXII based indicators of compromise," 2015.

[32] R. J. Joyce, D. Amlani, C. Nicholas, and E. Raff, "MOTIF: A malware reference dataset with ground truth family labels," *Computers & Security*, vol. 124, p. 102921, 2023.

[33] C. Julie, D. Mark, and S. Charles, "The trusted automated exchange of indicator information (TAXII)," *MITRE Corporation*, pp. 1–20, 2014.

[34] Z. Jun, Y. Qiben, L. Jianxin, S. Minglai, H. Zuti, and L. Bo, "TIMiner: Automatically extracting and analyzing categorized cyber threat intelligence from social data," *Computers & Security*, vol. 95, p. 101867, 2020.

[35] M. K. and S. Khandelwal, "The hacker news," 2010, access date: 15 May 2023. [Online]. Available: https://thehackernews.com/

[36] 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*, 2018, pp. 1–5.

[37] Z. Li, Q. A. Chen, C. Xiong, Y. Chen, T. Zhu, and H. Yang, "Effective and light-weight deobfuscation and semantic-aware attack detection for powershell scripts," in *Proceedings of the 26th ACM SIGSAC Conference on Computer and Communications Security*, 2019, pp. 1831–1847.

[38] Y. Lin, R. Liu, D. M. Divakaran, J. Y. Ng, Q. Z. Chan, Y. Lu, Y. Si, F. Zhang, and J. S. Dong, "Phishpedia: A hybrid deep learning based approach to visually identify phishing webpages," in *Proceedings of the 30th USENIX Security Symposium*, 2021, pp. 3793–3810.

[39] Malware Must Die, "Mmd-0056-2016 - linux/mirai, how an old elf malcode is recycled," 2016, access date: 15 May 2023. [Online]. Available: https://blog.malwaremustdie.org/2016/08/mmd-0056-2016-linuxmirai-just.html

[40] Y. Merah and T. Kenaza, "Ontology-based cyber risk monitoring using cyber threat intelligence," in *Proceedings of the 16th International Conference on Availability, Reliability and Security*, 2021, pp. 1–8.

[41] L. Metcalf and J. M. Spring, "Blacklist ecosystem analysis: Spanning jan 2012 to jun 2014," in *Proceedings of the 2nd ACM Workshop on Information Sharing and Collaborative Security*, 2015, pp. 13–22.

[42] Microsoft, "Threat intelligence integration in microsoft sentinel," 2023, access date: 12 May 2022. [Online]. Available: https://github.com/MicrosoftDocs/azure-docs/blob/main/articles/sentinel/threat-intelligence-integration.md

[43] MITRE Corporation, "STIX/TAXII Supporters List," 2014, access date: 23 November 2021. [Online]. Available: https://stixproject.github.io/supporters/

[44] S. Mohurle and M. Patil, "A brief study of wannacry threat: Ransomware attack 2017," *International Journal of Advanced Research in Computer Science*, vol. 8, no. 5, pp. 1938–1940, 2017.

[45] M. A. Napierala, "What is the bonferroni correction?" *Aaos Now*, pp. 40–41, 2012.

[46] OPSWAT, "MetaDefender Cloud-Advanced threat prevention and detection," 2012, access date: 27 May 2022. [Online]. Available: https://metadefender.opswat.com/

[47] A. Otis, B. Misha, and S. Jacob, "Mitre att&ck® for industrial control systems: Design and philosophy," *Mitre Corporation*, 2020.

[48] P. S. PandaLabs, "One third of all computer viruses that exist were created in the first 10 months of 2010," 2010, access date: 15 May 2023. [Online]. Available: https://www.pandasecurity.com/en/mediacenter/press-releases/one-third-of-all-computer-viruses-that-exist-were-created-in-the-first-10-months-of-2010/

[49] K. Paris, C. Thanasis, A. Sofia, S. Spiros, and T. Christos, "INTIME: A machine learning-based framework for gathering and leveraging web data to cyber-threat intelligence," *Electronics*, vol. 10, p. 818, 2021.

[50] Y. Park and T. Lee, "Full-stack information extraction system for cybersecurity intelligence," in *Proceedings of the 27th Conference on Empirical Methods in Natural Language Processing: Industry Track*, 2022, pp. 531–539.

[51] J. Pearl, *Causality*. Cambridge university press, 2009.

[52] Ponemon Institute, "Live threat intelligence impact report 2013," 2013, access date: 15 May 2023. [Online]. Available: https://www.ten-inc.com/presentations/Norse-Ponemon-Report.pdf

[53] ——, “The value of threat intelligence: Annual study of North American & United Kingdom companies,” 2019.

[54] G. Ratnam, “Cleaning up solarwinds hack may cost as much as $100 billion,” 2021, access date: 28 July 2022. [Online]. Available: https://rollcall.com/2021/01/11/cleaning-up-solarwinds-hack-may-cost-as-much-as-100-billion/

[55] A. Salem, S. Banescu, and A. Pretschner, “Maat: Automatically analyzing virustotal for accurate labeling and effective malware detection,” *ACM Transactions on Privacy and Security*, vol. 24, no. 4, pp. 1–35, 2021.

[56] C. Sauerwein, C. Sillaber, A. Mussmann, and R. Breu, “Threat intelligence sharing platforms: An exploratory study of software vendors and research perspectives,” 2017.

[57] B. Sean, “Standardizing cyber threat intelligence information with the structured threat information expression (STIX),” *Mitre Corporation*, vol. 11, pp. 1–22, 2012.

[58] M. Sebastián, R. Rivera, P. Kotzias, and J. Caballero, “Avclass: A tool for massive malware labeling,” in *Proceedings of the 19th International Symposium on Research in Attacks, Intrusions and Defenses*, 2016, pp. 230–253.

[59] S. Sebastián and J. Caballero, “Avclass2: Massive malware tag extraction from av labels,” in *Proceedings of the 36th Annual Computer Security Applications Conference*, 2020, pp. 42–53.

[60] Securelist by Kaspersky, “Wannacry and lazarus group – the missing link?” 2017, access date: 15 May 2023. [Online]. Available: https://securelist.com/wannacry-and-lazarus-group-the-missing-link/78431/

[61] Securelist by Kaspersky, “Wannacry ransomware used in widespread attacks all over the world,” 2017, access date: 15 May 2023. [Online]. Available: https://securelist.com/wannacry-ransomware-used-in-widespread-attacks-all-over-the-world/78351/

[62] A. Seth, “Granger causality,” *Scholarpedia*, vol. 2, p. 1667, 2007.

[63] F. Shota, K. Nobutaka, S. Tomohiro, and Y. Toshihiro, “CyNER: Information extraction from unstructured text of cti sources with non-contextual iocs,” in *Proceedings of the 17th International Workshop on Security*. Springer, 2022, pp. 85–104.

[64] Z. Shuofei, S. Jianjun, Y. Limin, Q. Boqin, Z. Ziyi, S. Linhai, and W. Gang, “Measuring and modeling the label dynamics of online anti-malware engines,” in *Proceedings of the 29th USENIX Security Symposium*, 2020, pp. 2361–2378.

[65] SIEMENS, “Caught in the crosschairs: Are utilities keeping up with the industrial cyber threat?” 2019, access date: 28 July 2022. [Online]. Available: https://assets.siemens-energy.com/siemens/assets/api/uuid:c723efb9-847f-4a33-9afa-8a097d81ae19/siemens-cybersecurity.pdf

[66] H. Slatman, “Awesome-threat-intelligence,” 2015, access date: 23 November 2021. [Online]. Available: https://github.com/hslatman/awesome-threat-intelligence/

[67] M. R. Smith, N. T. Johnson, J. B. Ingram, A. J. Carbajal, B. I. Haus, E. Domschot, R. Ramyaa, C. C. Lamb, S. J. Verzi, and W. P. Kegelmeyer, “Mind the gap: On bridging the semantic gap between machine learning and malware analysis,” in *Proceedings of the 13th ACM Workshop on Artificial Intelligence and Security*, 2020, pp. 49–60.

[68] K. Thomas, R. Amira, A. Ben-Yoash, O. Folger, A. Hardon, A. Berger, E. Bursztein, and M. Bailey, “The abuse sharing economy: Understanding the limits of threat exchanges,” in *Proceedings of the 19th International Symposium on Research in Attacks, Intrusions and Defenses*. Springer, 2016, pp. 143–164.

[69] W. Tounsi and H. Rais, “A survey on technical threat intelligence in the age of sophisticated cyber attacks,” *Computers & security*, vol. 72, pp. 212–233, 2018.

[70] X. Ugarte-Pedrero, M. Graziano, and D. Balzarotti, “A close look at a daily dataset of malware samples,” *ACM Transactions on Privacy and Security (TOPS)*, vol. 22, no. 1, pp. 1–30, 2019.

[71] VirusTotal, “VirusTotal-Free Online Virus, Malware and URL scanner,” 2012, access date: 27 May 2022. [Online]. Available: https://www.virustotal.com

[72] T. Vissers, J. Spooren, P. Agten, D. Jumpertz, P. Janssen, M. Van Wesemael, F. Piessens, W. Joosen, and L. Desmet, “Exploring the ecosystem of malicious domain registrations in the. eu tld,” in *Proceedings of the 20th International Symposium on Research in Attacks, Intrusions and Defenses*. Springer, 2017, pp. 472–493.

[73] B. Xander, L. P. Victor, F. Pawel, V. G. Tom, G. C. H, M. Giovane, T. Samaneh, J. Wouter, and van Eeten Michel, “Helping hands: Measuring the impact of a large threat intelligence sharing community,” in *Proceedings of the 31st USENIX Security Symposium*, 2022.

Appendix A
Artifact Appendix

This document describes how to use the source code of CTI-Lense, a tool that collects STIX data from a set of open CTI sources and systematically analyzes the gathered data. The code evaluates the STIX dataset in terms of volume, timeliness, diversity, and quality.

A. Description & Requirements

Below we explain about how to set up CTI-Lense, providing links to get the source code, dataset, and docker image. The docker image includes all requirements for our experiments, such as software, source code, and dataset. So, we highly recommend you to directly use our docker image.

1) Access:

- Source code (Github): https://github.com/SKKU-SecLab/CTI_Lense.
- Source code (figshare): https://dx.doi.org/10.6084/m9.figshare.24581004.
- Dataset (figshare): https://dx.doi.org/10.6084/m9.figshare.24126336.
- Docker image: jinbumjin/cti-lense:artifactv1.1

2) Hardware Dependencies: Our experiments are run on a virtual machine with at least 4 cores and 32 GiB memory.

3) Software Dependencies: We provide the environment for our experiments with source code using a docker image, so you must install the docker client.

4) Benchmarks: None.

B. Artifact Installation & Configuration

To begin with, the docker client should be installed on Ubuntu 18.04. Then, download and run the docker image of CTI-Lense. The following commands will install the required dependencies and pull/run the docker image:

```
$ sudo apt-get install docker.io
$ sudo docker pull jinbumjin/cti-lense:artifactv1.1
$ sudo docker run -it jinbumjin/cti-lense:artifactv1
.1 /bin/bash
```

C. Experiment Workflow

To reproduce the results that we report in our paper, follow the instructions below:

First, start the MongoDB service in the docker container. It takes up to 10 minutes to enable the MongoDB service. Then, to get our experimental results, simply run `CTI_Lense.py` code with the following commands in the docker container.

```
/CTI_Lense# service mongodb start
/CTI_Lense# python3 CTI_Lense.py -e analysis_type
```

D. Major Claims

Our major claims are as follows:

- (C1): CTI-Lense shows that STIX data producers actively generate and disseminate STIX data within 2–12 days after a security incident. This claim is supported by Experiment 1 (E1), whose findings are detailed in Section 5.
- (C2): CTI-Lense shows that STIX data is mostly based on *Indicator* objects, accounting for over 90% of all the data. Most of the *Indicator* objects contain simple indicators of compromise information, such as malicious file hash or URL strings. This is evidenced by Experiment 2 (E2), whose results are reported in Table IV and Table V of Section VI.
- (C3): CTI-Lense shows that producers do not always adhere to the intended use of objects and attributes in the STIX standard. First, STIX does not always contain accurate values, although it generally includes verified values for file objects and URIs. Second, producers do not consistently select the appropriate objects and attributes to represent values. Instead, they often describe specific information in a narrative way, bypassing specific STIX attributes. This is evidenced by Experiment 3 (E3), whose results are reported in Figures 9, 10, 12, and Table VI and VIII of Section VII.

E. Evaluation

The operational steps and experiments for our evaluation are as follows:

1) Experiment 1 (E1): [Timeliness] [1 compute-second]: This experiment shows the causality test results between daily security incidents from Malpedia and the daily shared STIX data. You can observe that STIX data are disseminated 2—12 days post the incident reporting on Malpedia, with a significance level $p < 0.05$.

[Preparation] Ensure that the docker container is running and the mongodb is enabled in the docker container.

[Execution] Run the "CTI_Lense.py" script to obtain our experimental results using the command:

```
/CTI_Lense# python3 CTI_Lense.py -e timeliness
```

[Results] The output shows the results of causality tests between the daily Malpedia and STIX datasets, for time lags of 1 to 30 days. The p-values for time lags of 2 to 12 days are less than 0.0008 (adjusted by Bonferroni correction).

2) Experiment 2 (E2): [Diversity] [3 compute-minutes]: This experiment shows how STIX data objects and attributes are utilized, as detailed in Table IV of Section VI. It also elucidates the attributes used in *Indicator* objects, as shown in Table V of Section VI.

[Preparation] Ensure that the docker container is running and the mongodb is enabled in the docker container.

[Execution] Run the "CTI_Lense.py" script to obtain our experimental results using the command:

```
/CTI_Lense# python3 CTI_Lense.py -e diversity
```

[Results] The output shows the results of Table IV and V. Table IV indicates that the *Indicator* object type is most frequently employed, accounting for 98.77% in STIX 1 and 94.93% in STIX 2. Table V indicates that a considerable number of *Indicator* objects do not contain advanced attributes. Specifically, in STIX 1, only 53.73% of the objects have the attribute *Type*, 34.76% have *Indicated_TTP*, and a mere 0.09% have *Test_Mechanisms*. In STIX 2, only 17.13% of the objects have both *Indicator_types* and *Kill_chain_phases*.

3) Experiment 3 (E3): [Quality] [1 compute-hour]: This experiment measures and reports the quality of STIX data, including improper values and usages, in Figures 9, 10, 12, and Table VI and VIII of Section VII

[Preparation] Ensure that the docker container is running and the mongodb is enabled in the docker container.

[Execution] Run the "CTI_Lense.py" script to obtain our experimental results using the command:

```
/CTI_Lense# python3 CTI_Lense.py -e quality
```

[Results] The output shows the results of Figures 9, 10, 12, and Table VI and VIII. These results indicate that STIX data often contains inaccurate values and narrative descriptions instead of specific STIX attributes.