Journal of Information Security and Applications 58 (2021) 102715

Contents lists available at ScienceDirect

Journal of Information Security and Applications

journal homepage: www.elsevier.com/locate/jisa

ETIP: An Enriched Threat Intelligence Platform for improving OSINT correlation, analysis, visualization and sharing capabilities

Gustavo González-Granadillo [a,*], Mario Faiella [a], Ibéria Medeiros [b], Rui Azevedo [b], Susana González-Zarzosa [a]

[a] Atos Research and Innovation, ATOS, Spain
[b] LASIGE, Faculty of Sciences, University of Lisboa, Portugal

ARTICLE INFO

Keywords:
Threat Intelligence Platforms
Open Source Intelligence (OSINT)
Data enrichment
Threat assessment

ABSTRACT

Open Source Intelligence (OSINT) data is collected by publicly available sources to be used by intelligence contexts among which Threat Intelligence Platforms (TIPs) are the main consumers. These platforms help organizations aggregate, correlate, and analyze threat data from multiple sources in real-time to support defensive actions. However, considering the unstructured nature of the collected data, TIPs require the data to be correlated with real-time information coming from the monitored infrastructure, before being further analyzed and shared. This paper presents *ETIP*, an *Enriched Threat Intelligence Platform* with extended capabilities in terms of import, quality assessment processes, visualization and information sharing in current TIPs. The platform receives structured cyber threat information from multiple sources and performs the correlation among them with static and dynamic data coming from external sources and the monitored infrastructure. This allows the evaluation of a threat score through heuristic-based analysis, used to enrich the information received from OSINT and other sources. The final result is sent to external entities, such as SIEMs, to be further used for a more in-depth analysis, and to be shared with trusted organizations.

1. Introduction

The number and impact of cyber attacks have drastically increased during the last years, as revealed by reports written by governments and companies, especially in terms of how much these threats could harm them from an economic point of view. The Council of the Economic Advisers of the United States [1] estimated that malicious cyber activity had an impact in the U.S. economy between 57 billion and 109 billion dollars in 2016. Cybersecurity Ventures [2] identified cyber crime as the *"greatest threat to every company in the world"*, predicting that it will cost the world more than six trillion dollars annually by 2021. Moreover, the global management consulting firm Accenture [3], during a study conducted in 2017 affirmed that cyber crime, on an annual average, is costing organizations 11.7 million dollars (around 23 percent more than the previous year). These successful incursions potentially allow groups of attackers to acquire valuable intellectual properties and secrets. With the aim of facing these menaces to protect precious internal and sensitive data as well as critical assets, it is crucial to have timely access to relevant and accurate information about them.

Collecting and processing Open Source Intelligence (OSINT) information is becoming a fundamental approach for obtaining cybersecurity threat awareness. Recently, the research community has demonstrated that useful information and Indicators of Compromise (IoC) can be obtained from OSINT [4,5]. Research studies have provided evidence that useful and early information can be obtained from social networks e.g., Twitter [5,6]. Twitter is a useful OSINT data source that aggregates timely data from multiple sources which is simple to process and analyze. Given the fact that users regularly tweet about their activities and unusual events found, it is possible to obtain valuable security-related data from Twitter before it becomes available on public databases (e.g., NDV, ExploitDB, etc.) [7,8].

Besides the research oriented efforts, all Security Operation Centre (SOC) analysts get updated about new threats against their IT infrastructures by collecting and analyzing cybersecurity OSINT data. Nevertheless, skimming through various news feeds is a time-consuming task for any security analyst.

Furthermore, analysts are not guaranteed to find news relevant to the IT infrastructure they oversee. Tools are therefore required, not only to collect OSINT, but also to process it, aiming at enhancing the quality

[*] Corresponding author.
E-mail address: gustavo.gonzalez@atos.net (G. González-Granadillo).

https://doi.org/10.1016/j.jisa.2020.102715

of the information carried on OSINT to SOC analysts, for instance, to benefit from the potential they have. In addition, such tools must filter only the relevant parts for the SOC analysts, thus decreasing the amount of information and consequently, the time required to analyze it and act upon. When appropriate, the filtered information must be further processed to extract IoCs.

Moreover, a proper quality assessment is needed, to check if gathered data can be considered as valuable Threat Intelligence (denoted as TI). Sillaber et al. [9] identified TI quality evaluation as one of the main challenges in actual cybersecurity information sharing scenarios, mainly caused by the limitation of existing TI sharing tools, as well as the lack of suitable and globally recognized standards and ontologies [10]. These assessment processes can provide more insights for inferring the impact that some cyber attacks could have with respect to internal assets and resources, prioritizing threat detection and incident response.

In addition, the ability to share OSINT information is often not enough. TI must be expressed, and then, shared using specific standards, allowing involved parties to speed up processing and analysis phases of received information, achieving interoperability among them.

This paper is an extended version of our previous work [11] where we introduced the *Enriched Threat Intelligence Platform* (denoted as *ETIP*), aiming at extending import and information sharing capabilities of internal detection and monitoring systems (e.g., SIEMs) and also improving the quality assessment of received cybersecurity events. This paper provides detailed information about the heuristics used in the computation of the threat score, the evaluated features, as well as the selected attributes and their individual scores. In addition, this article provides information about the visualization platform used to display the network topology in a graphical manner and to present results obtained by the tool (including the threat score).

The final objective is to integrate the relevant security data coming from public sources (e.g., social networks, OSINT), after going through a quality information enhancing process, with data gathered from the monitored infrastructure through specific detection and monitoring systems (e.g., SIEMs, IDS, IPS), to anticipate and improve threat detection and incident response. This integration has been defined as a crucial activity in order to produce real and valuable TI [12]. In this context, on the one hand, it arises the need of a component that relates and aggregates collected OSINT data, generating thus new enriched data. On the other hand, it also requires a component that considers potential security issues in the monitored infrastructure to be correlated with the received OSINT data, providing a threat score that helps to identify its relevance and priority.

The threat score will complement the usage of static information about the monitored infrastructure with dynamic and real-time threat information reported from inside the network in the way of IoCs. This dynamic evaluation is based on heuristic analysis which allows determining the priority of the incoming OSINT data, by assigning a threat score to it. The produced object integrating the information received from OSINT data sources through its calculated threat score is sent directly to other security systems and tools (e.g., SIEMs) for visualization, storage, processing, or feedback, and could optionally be shared with external trusted organizations.

Both, original and enriched data are shared with trusted external parties, improving collaborations among different organizations (that may belong to different sectors e.g., finance, health, energy, public administration). Ardieta et al. [13] stated that tasks like threat detection and incident response could not be handled in an isolated way, from a single organization point of view, highlighting the crucial role of threat information sharing, which has also been considered as one of the five critical activities for enhancing defense capabilities [14].

The contributions of this article are summarized as follows:

- The *ETIP* platform that extends import and information sharing capabilities of internal detection and monitoring systems (e.g., SIEMs);
- A method that improves the quality assessment of received cybersecurity events;
- A process that integrates relevant security data coming from public sources with data gathered from the infrastructure through specific detection and monitoring systems;
- A dynamic heuristic-based analysis to calculate the threat score of each processed Indicator of Compromise (IoC) ;
- The deployment of the platform over a real attack scenario.

The remainder of this paper is structured as follows: Section 2 presents related work regarding Threat Intelligence Platforms. Section 3 describes the architecture of our proposed Enriched Threat Intelligence Platform (*ETIP*). Section 4 details the Threat Score Evaluation process. Section 5 illustrates the applicability of our approach with a use case scenario and the obtained results. Section 6 discusses and analyzes the main results of this article. Finally, conclusions and perspective for future work are presented in Section 7.

2. Related work

Several standard formats have been proposed to facilitate cyber intelligence sharing among platforms. Examples of such formats are the Open Indicators of Compromise (OpenIoC [15]), Structured Threat Information eXpression (STIX [16]), Trusted Automated eXchange of Indicator Information (TAXII [17]).

Besides the great variety of commercial and open-source security data analytic platforms, current tools are unable to cope with the new and complex attack patterns. Most of them lack of capabilities to collect, process, store and use Open Source Intelligence (OSINT) data to identify, visualize and prioritize threats [18–21].

Some studies have identified Threat Intelligence Platforms (TIPs) as ideal tools for data collection, storage, sharing, and integration with external entities (e.g., other security platforms and tools, as well as specific groups for handling incident response and threat management such as SOCs, CERTs, CSIRTs). Several TIPs are available in the market (most of them under commercial license). In terms of open-source solutions, Tounsi and Rais [22] provides a survey including (i) the Malware Information Sharing Platform (MISP) [23], (ii) the Collective Intelligence Framework (CIF) [24], (iii) the Collaborative Research Into Threats (CRITs) [25], and (iv) Soltra Edge [26] (only a limited version is available with this kind of license).

Sauerwein et al. [27], provide an exploratory study of software vendors and research perspectives of threat intelligence sharing platform, and conclude that the market for threat intelligence sharing is still developing. Moreover, ENISA provides an updated report about opportunities and limitations of actual TIPs [28], suggesting guidelines to overcome them.

Owen [29] proposes Moat, a powerful tool that covers known bad actors and consume data from multiple sources such as vulnerability systems and port scanners. Moat has been integrated with SIEMs using STIX and XML formats for sharing purposes but it is not yet defined for other well-known standards such as TAXII.

Some commercial SIEMs (e.g., LogRhythm [30]) have added security intelligence to its analytic platform. Their approach uses rich context enabled by threat intelligence from STIX/TAXII-compliant providers, commercial and open-source feeds, as well as internal honeypots. As a result, the platform uses these data to reduce false-positives, detect hidden threats, and prioritize concerning alarms.

Many companies started relying on TIPs for overcoming gaps and limitations of actual detection and monitoring systems, especially SIEMs [31]. They are in charge of retrieving structured and unstructured data from diverse external sources, and perform various complex operations, such as filtering, aggregation, normalization, detection, analysis and enrichment, as well as the injection of results into SIEMs. However, their implementation and usage are still in their infancy and, as stated in [27], many drawbacks have to be addressed, for instance,

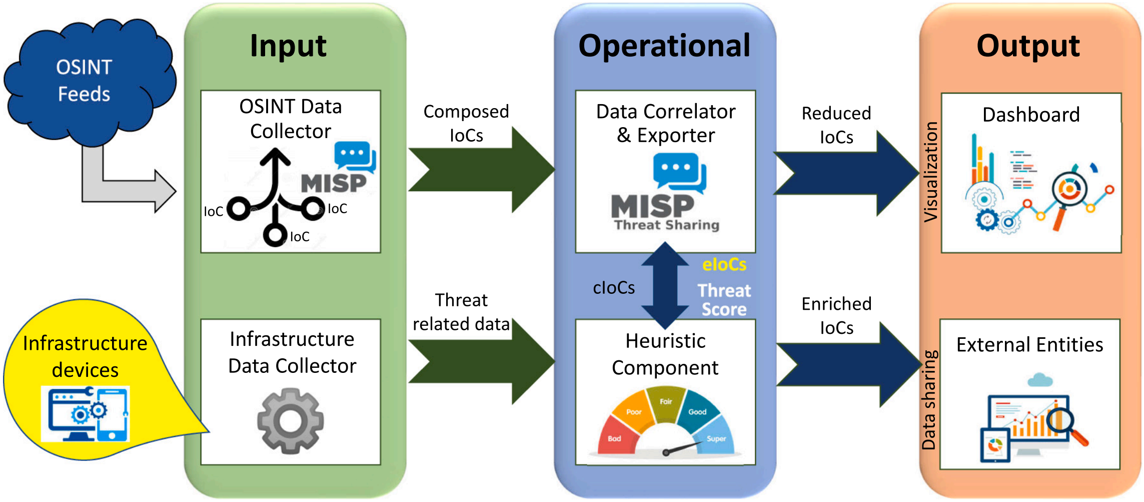


Fig. 1. The Enriched Threat Intelligence Platform architecture.

the dynamic trust assessment of external sources and advanced analysis capabilities still require manual work to make the retrieved information effectively actionable.

In addition, previous works have provided evidence that useful and early information can be obtained from OSINT sources [5,6]. Several approaches [32–34] have been proposed to collect context-specific OSINT by using a keyword set. As a result, relevant tweets are selected and classified using machine learning techniques, making it possible to discover threat data in OSINT sources, before they are included in threat databases. Other approaches based on unstructured text use ontologies and/or machine learning techniques to detect and predict threat patterns [35–37]. However, most of these approaches require the development of OSINT tools to filter valuable information from the noise generated by OSINT feeds, and require validation from security analysts to discard misleading and inaccurate data.

To the best of our knowledge, more research is needed about TIPs, and their integration with other security tools. Our approach suggests the use of a platform for collecting and aggregating cybersecurity related information from OSINT, relying on MISP for storing and managing the resultant IoCs, which will be further enriched with a threat score, for prioritizing possible defense actions. The selection of MISP is due to its numerous advantages, e.g., its ability to be integrated with SIEMs and IDSs; its high flexibility features to integrate internal and custom solutions; the support of specific data exchange standard, such as STIX, as well as good built-in information sharing capabilities; the availability of a very detailed on-line documentation [38]; and a huge and responsive on-line community, in case of development issues. The outcome of this platform will feed systems, like SIEMs and IDSs, with actionable information that will improve the detection of cyber threats, and could easily be shared, in an automated way, with internal SOCs, CERTs and CSIRTs, as well as with other trusted organizations.

3. Enriched Threat Intelligence Platform (*ETIP*)

The platform we propose in this paper, called Enriched Threat Intelligence Platform (*ETIP*), involves generating enriched threat intelligence, leveraging from OSINT data and data provided by external sources and organization's IT infrastructure (e.g., firewalls, IDS, IPS) which are correlated, evaluated and represented as a threat score. This enriched information can be integrated by defense mechanisms to prevent attacks against the organization and, hence, combat cybercrime. Also, it can be visualized graphically for better understating and analyzing its interconnections and relevant data. This section gives an overview of the approach and the architecture of the platform, for which definitions of the main elements used in *ETIP* are presented, and the characteristics of each module is provided.

3.1. Overview

ETIP's architecture is presented in Fig. 1, which illustrates the three main modules it comprises, namely: (i) *Input Module* that includes the IoC generators from OSINT, as well as infrastructure tools and devices that aggregate threat-related data; (ii) *Operational Module* responsible for deploying the heuristic analysis process to calculate the threat score of the data collected and correlated and storing them for later usages; (iii) *Output Module* that contains the tool dashboard to visualize the enriched information generated and connections to security data analytic platforms (e.g., SIEMs), allowing the exportation of the enriched data to such platforms.

The first module collects security events (i.e., IoCs) provided from different OSINT feeds as well as infrastructure data. IoCs are processed and analyzed, resulting in IoCs with more information (i.e., *composed IoCs–cIoCs*). The second module receives these composed IoCs and correlates them with information collected from the infrastructure (e.g., IP addresses used, open ports, protocols in use, etc.), generally present in logs generated from security devices (e.g., IDS, Firewalls). Both, cIoCs and infrastructure data are contrasted to identify if there is a match. In such a case, the threat score of the matched cIoC will be increased by the heuristic component, denoting that cIoC carries potential threat data for organization's devices. For example, if a cIoC indicates there is a new vulnerability affecting Windows 10 machines, and we detect at least one active device in the target infrastructure running Windows 10, then the threat score is increased for this particular cIoC. Applying a heuristic analysis to these data, the resulting IoC can be further transformed into an *enriched IoC (i.e., eIoCs)*, providing more insights about how much the incoming information could be considered as real intelligence by the enterprise. The third module may use the eIoCs or a reduced version of them (i.e., *reduced IoCs–rIoCs*) for sharing and visualization purposes.

3.1.1. Main key concepts

This section provides definitions of key concepts that are widely used during the introduction the proposed solution and main outcomes generated.

- **Indicator of Compromise (IoC):** Denoted also as sIoC (single IoC) are pieces of forensic data, such as data found in system log entries or files, that identify potentially malicious activity on a system or network [39]. sIoCs are useful in detecting data breaches, malware infections, or other threat activity.

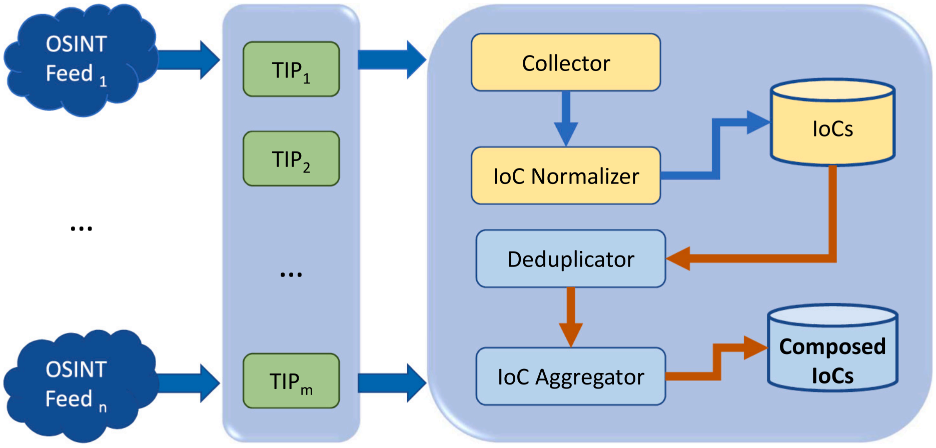


Fig. 2. OSINT data collector architecture.

- **Composed Indicator of Compromise (cIoC):** A cIoC is the result of the aggregation, interrelation and normalization of OSINT data regarding a same threat, which is retrieved from various source feeds and can be expressed in different formats (e.g., plaintext, csv).
- **Enriched Indicator of Compromise (eIoC):** An eIoC is the enriched version of a cIoC, obtained after the correlation of the latter with static and real-time information associated with the monitored infrastructure. The result of this process is a threat score (detailed later) that will be added to cIoC, enhancing and transforming it in an eIoC. For this reason the word "enriched" has been used.
- **Reduced Indicator of Compromise (rIoC):** An rIoC is the reduced version of the corresponding enriched one. The latter could potentially contain a huge amount of information, not worthy to be visualized, but still useful for future analysis and correlation tasks. Therefore, only the rIoC, with just the most relevant information from the monitored infrastructure point of view, will be sent to the dashboard, while the eIoC will be stored locally, or shared with external entities.

3.2. *Input module*

This module is composed of two elements in charge of collecting data coming from OSINT sources and the infrastructure and a MISP instance to centralize all collected and generated data. Its main objective is to collect, clean, and aggregate data to feed the operational module with composed Indicators of Compromise (cIoCs) and other threat related data for further data processing and analysis. The remainder of this section details the components of this module.

3.2.1. *OSINT data collector*

This component aims to generate cIoCs based on the aggregation of OSINT data. The architecture of this component is represented in Fig. 2 and a detailed version of it can be found in [40]. The main parts are the following:

- *OSINT Feeds.* The component is configured with different types of Open Source Intelligence (OSINT) feeds about security events (e.g., cyber-attacks, malware domains, vulnerability exploitation, IP blacklists) provided by several sources, such as free and collaborative organizations.
- *TIPs.* Different TIPs are used in parallel to collect several OSINT data provided from diverse feeds, which take advantage of the enrichment capabilities they offer, such as improving OSINT threat intelligence with external data not included in OSINT feeds (e.g., asn source, whois).
- *Collector.* The output of the different TIPs, as a form of IoCs, is channeled to a collector module configured in MISP for the effect. The TIP's IoCs are seen as OSINT feeds but in an IoC format (e.g., STIX, MISP format).
- *IoC Normalizer.* Since IoCs might be collected in different formats (depending on the format adopted by TIPs), it is necessary to normalize them in a single and common format (e.g., MISP format [23], STIX, etc.). After this process, they are stored in a database to be processed by the component. MISP performs this task, i.e., receives IoCs in different formats and normalize them, representing them in MISP format.
- *Deduplicator.* IoCs received from different TIPs can be equal, since TIPs can be configured with the same OSINT feeds. The deduplicator module analyzes the received IoCs with those already existing in the MISP database with the aim to identify duplicated IoCs and remove them before being processed by the IoC aggregator module. After an IoC is normalized and before being stored in the database, the deduplicator uses a metric of similarity, called *contained similarity* [40], to infer the existence of duplicates, combining the normalized IoC with those in the database and calculating the similarity between each pair of combined IoCs. When resulting similarities equal to one, (e.g., in a pair of IoCs both are equals or one IoC is contained in the other IoC), it means that the deduplicator found duplicates, and then, it discards them.
- *IoC Aggregator.* Aggregates different but related IoCs, and generates new ones. The process consists on identifying IoCs that contain relevant interrelated information, aggregating them in a same set, and then, merging that information into a single IoC, creating a new IoC that we call *composed IoC*. These new IoCs are stored in the database for later be used by the threat intelligence sharing component (see Section 3.3).

Given the relevance of the *IoC Aggregator*, as it is the main part of the OSINT data collector component and performs the aggregation, correlation and representation steps, next we give more details about it. The process performed by the *IoC Aggregator* starts by searching for correlations between the different IoCs stored in the database. In other words, it searches for those pairs of IoCs where its similarity ranges $[0, 1]$, i.e., there is common (intersection of) data within a pair. Once correlations have been identified, it generates new IoCs composed of the correlated IoCs. It performs its function in two steps: (1) *aggregation*, it queries the database to identify IoCs that contain relevant related information fields in order to determine sets of related IoCs; (2) *representation*, for each resulting set, it merges the information contained in different IoCs into a single one, eliminating duplicated attributes, and stores the new composed IoC for later use.

To establish the correlations the component resorts of one of two correlation methods we defined, named *naive* and *deeper*, that would allow the identification of groups of correlated IoCs, generating thus the sets of IoCs. However, before performing any method, an initial filtering stage is applied to identify only IoCs that respect specific rules, i.e., by eliminating events that will bring no added value (such as blacked IP lists), this allows to create a *subset of IoCs of interest*. This filter is based on a previous configuration of the component, in which we can define the level of detail about threats we want to process, i.e., the level of information that IoCs carry. In the naive method only direct correlations are identified, in the sense that the composed IoC is built from a central IoC and all those IoCs that share one or more attributes with it. This means that in the naive method each IoC of the subset of IoCs of interest is considered a central IoC and for each of them are identified the direct correlations. The resulting composed IoCs can overlap each other. On the other hand, the deeper method creates a graph with all events in the IoCs set, where each IoC is a node and the edges represent shared attributes between IoCs, and sets interconnecting IoCs are identified as a source for a new composed IoC.

3.2.2. Infrastructure data collector

Unlike the OSINT data collector, this component obtains information related to the monitored infrastructure that could lead to internal indicators of compromise (e.g., hashes, signatures, IPs, domains, URLs, etc.). This information can be obtained from the system log files that record events occurring in an operating system or messages between different users of a communication software. Event logs, system logs, server logs, Web logs, and application logs, are examples of the infrastructure input data. These data can be collected from a variety of devices (e.g., firewalls, intrusion detection and prevention systems, honeypots, and other security sensors) that could provide indications of malicious activities in the system.

In addition, this component gathers information of internal monitoring devices and operations from the infrastructure (e.g., installed applications, operating systems, threat actors, intrusion tools, vulnerabilities, etc.) that will be contrasted with the data coming form external sources in order to assess their corresponding risk level. This correlation process, between information received from external sources and cybersecurity related data detected with internal security tools, has been defined as a critical activity for obtaining relevant and actionable threat intelligence [12].

Concretely speaking, if the infrastructure data collector detects an application running on Windows XP, for which a given vulnerability has been recently detected by the OSINT data collector, the correlation of these data coming from these two sources will indicate a potential attack scenario, for which the threat score must be increased accordingly.

3.2.3. Generating composed IoCs from OSINT and infrastructure data

Both OSINT and infrastructure data components have been built considering the MISP platform. The *deduplicator* and *IoC aggregator* modules from the OSINT data collector were developed in Python 3 and integrated in MISP. MISP acts as the *collector* and *IoC normalizer* to receive and normalize the data that can be provided from different TIPs (in case of OSINT data), or from organization's infrastructure (in case of infrastructure data), and then, it uses the other two modules to process the received data, respectively, to eliminate duplicated IoCs and aggregate IoCs associated with the same threat category in a single IoC (generating composed IoCs).

The OSINT data collector offers to the end-user the possibility of configuration based on two criteria: (1) the trust level of the IoC assigned by the MISP community, where, for example, an IoC with level 2 means that IoC has the trustiest level of confidence and its information is relevant; (2) the interrelation type between IoCs which will be considered by the *IoC aggregator* module. This interrelation can be based either on the IoC as a whole, or on their attributes. The former only allows interrelations between IoCs that belong to the same threat category or type of infrastructure event, whereas the latter permits a deeper analysis and connections among IoCs, allowing to generate new data provided by IoCs of different categories (e.g., IoCs belonging to the MISP *network* category and from type of *vulnerability*).

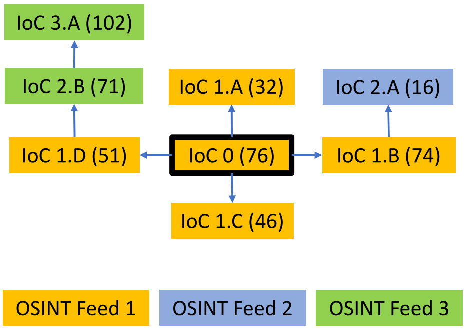


Fig. 3. Schematic for the creation of a composed IoC.

The input module considers dividing the OSINT feeds in two categories: (1) low level feeds, which consist mainly of IP addresses and URLs; and (2) high level feeds, which contain a more advanced analysis with information about network artifacts, campaigns, etc.; that feed TIPs (e.g., CRIT, MISP). It performs queries to the database to identify new entries and other entries that have matches, and then merge them forming a new IoC and inject it into the database, which is labeled with a tag that allows identifying it as a rich IoC and avoids the creation of loops.

Fig. 3 exemplifies a composed IoC formed from OSINT data. In the figure, starting from an IoC that contained 76 elements, we were able to identify 7 other IoCs, originating from 3 distinct OSINT feeds, that were correlated. The merging of these IoCs allowed the creation of a new IoC containing 468 elements.

The resulting composed IoCs (cIoC), regardless the source they provided (OSINT or infrastructure), are stored in the *Composed IoC* database which follows the MISP database structure since a cIoC has the same format of a MISP IoC plus some additional attributes created to distinguish the cIoC from the original IoCs. Later, they will be processed jointly by the operational module to find malicious activity in the organization TI infrastructure.

3.3. Operational module

This module is based on MISP, which is able to correlate static and real-time information (e.g., Indicators of Compromise), related to the monitored infrastructure, with data coming from external OSINT sources (represented as cIoCs) through OSINT data fusion and analysis tools, to check the relevance and accuracy of the data. The result of these actions is what we call the enriched IoC (eIoC), an IoC that combines both OSINT and infrastructure data regarding a same threat. Furthermore, the module is also able to evaluate the threat score of eIoCs and share both the original (infrastructure and OSINT data, and cIoCs) and the enriched information with external entities, in an automated way.

The proposed module architecture, depicted in Fig. 4, is composed of two main elements: (i) *Data Correlator & Exporter*, represented by a MISP instance and in charge of correlating data from both OSINT sources (i.e., composed IoCs) and internal sources (i.e., infrastructure data), as well as sending the *enriched IoCs* to internal components, systems and tools (e.g., SIEMs) or sharing them with trusted organizations; and (ii) the *Heuristic Component*, in charge of performing the heuristic analysis, with the final aim of computing a Threat Score, enriching the data coming from the Data Correlator, and sending it back to the MISP Instance. In Fig. 4, we include the two components of the input module for better illustrating and understanding the interconnection of both input and operational modules.

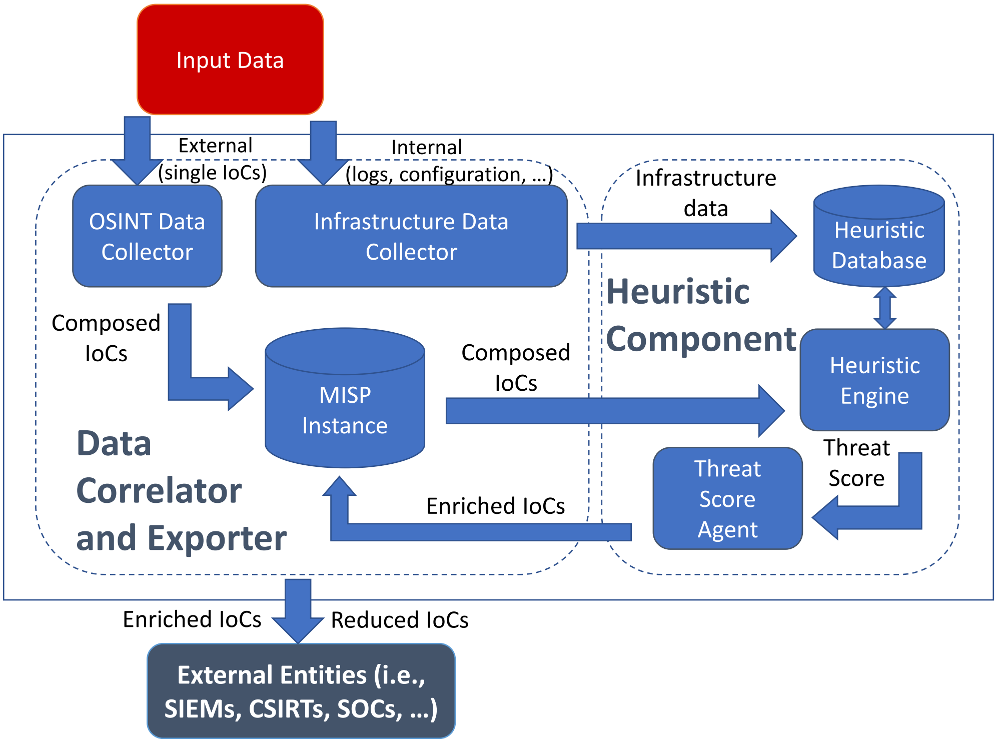


Fig. 4. Operational module architecture, including the OSINT and infrastructure data collectors.

3.3.1. Data correlator & exporter

From Fig. 4, the OSINT and Infrastructure data collectors are responsible of capturing useful data from OSINT, and the monitored infrastructure, and generating cIoCs in order to evaluate a set of predefined heuristics and to compute a threat score. Composed IoCs are stored in the MISP Instance Database, whereas collected infrastructure data are stored in the Heuristic Component Database.

The integration between security tools, as well as internal SOC and CERTs/CSIRTs, and the threat intelligence sharing module is possible thanks to the adoption of MISP. The objective is to use, as much as possible, the built-in sharing capabilities of the platform when this interaction takes place, such as a zeroMQ publish–subscribe model [41]. MISP comes with so-called “MISP-modules”, used both for ad-hoc import and export of threat information. If required, new modules could be created from scratch and integrated with the MISP Instance, without modifying the core functionalities of the platform. The deduplicator, aggregator IoC and heuristic components fit on these kind of modules, which were built from the scratch and integrated with MISP.

Data stored in the MISP instance is represented through JSON formats (e.g., STIX, MISP events), or through simple documents related to generic information. Since its usage is of great interest to the heuristic component, data can be also stored in a different way, using a private non-relational database such as MongoDB [42] (as presented in Section 7.4), which simplifies the information retrieval by the heuristic engine and allows for a full control of the analysis performed by the tool.

The adoption of MISP makes it possible to automatically share data with external entities thanks to its built-in information sharing capabilities. For those cases in which the external entity is using a MISP instance, the sharing process is performed by simply synchronizing both instances. Otherwise, MISP comes with a list of REST APIs, which are accessed from any internal and external services with different levels of access rights, to directly interact with its database, to push/pull cyber-security related events.

3.3.2. Heuristic component

The heuristic component receives information coming from multiple sources (e.g., OSINT data, infrastructure, IoCs, etc.) to be used in the Threat Score (TS) analysis performed by the heuristics engine. This latter considers a set of conditions that are evaluated for every single feature. A score is assigned to every feature (i.e., individual score). The sum of all individual scores results into the Threat Score associated with the data being analyzed.

Data could be dynamic (e.g., IoCs detected in the infrastructures) or static and generic information about a specific infrastructure (e.g., used sensors, operating systems, specific lists of IP addresses). The Threat Score Agent is responsible for the generation of the resulting enriched Indicator of Compromise (eIoC), including the Threat Score for the security information received from OSINT data sources. The eIoCs shared by this component includes the same information received from OSINT, as well as the associated Threat Score and the features considered in the evaluation. A detailed description of this component is provided in Section 4.

3.4. Output module

The Output module of our platform is mainly in charge of representing graphically the most relevant information contained in the eIoCs, as the form of reduced IoCs (rIoC) produced in the operational module. Enriched IoCs can contain a great number of features that can reduce efficacy of the visualization process. In order to avoid such limitations, a reduced IoC (rIoC), composed of information related to the infrastructure and the most relevant information of eIoC which is obtained from the attributes that support the threat score, is used for this purpose. The eIoC could be, instead, shared with other external entities, which could be both internal security tools or trusted organizations. The reminder of this section details each component of the output module.

3.4.1. Dashboard — graphical representation

The *Dashboard* provides a graphical representation of the infrastructure topology by highlighting the alarms and rIoCs associated to each node composing the infrastructure’s network, as depicted in Fig. 5.

Each node will have in its upper left side a circle indicating the number and severity of the alarms (in colors green, yellow and red),

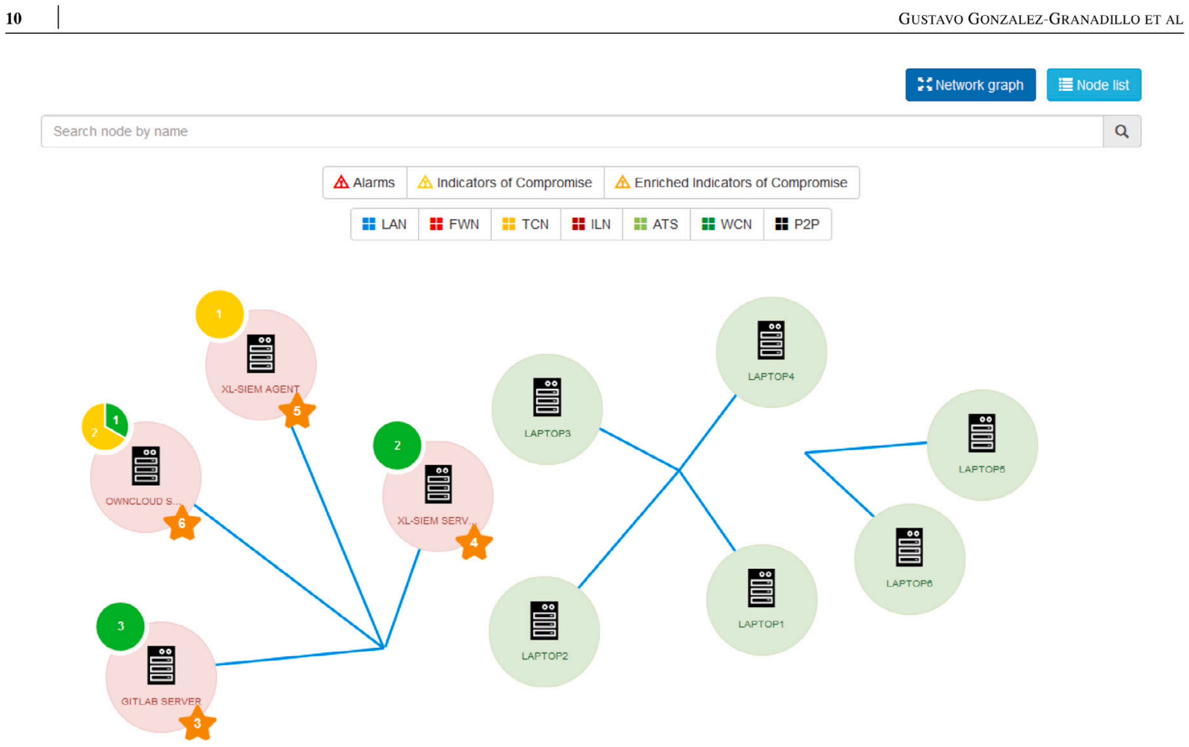


Fig. 5. *ETIP*'s Dashboard. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

and in its lower right side, a star indicating the number of rIoCs related to that particular node.

Alarms will indicate the number of issues, IP source and destination, as well as a brief description of the issue. rIoCs will indicate the number of detected vulnerabilities, the Common Vulnerability Exposure (CVE) [43], the associated threat score, a brief description of the vulnerability and the affected application. A system inventory containing the nodes, and their installed applications is required to perform the match.

In addition, the dashboard provides, in a separate tab, information about the type of node (e.g., Server, Workstation); the IP addresses (known, unknown, source, destination); the operating system (e.g., Linux, Windows); and the connected networks (e.g., LAN, WAN). The right side of the dashboard shows the nodes of the infrastructure with at least one security issue, for which, alarms and IoCs are provided.

Reduced IoCs are generated automatically by the platform after performing the threat score computation. Three attributes compose the rIoC: (i) information about the vulnerability (i.e., CVE number and description); (ii) threat score value (ranging from zero to five); and (iii) affected assets/applications in the infrastructure. The relevance of these data is validated by the security administrator based on expert knowledge and potential matches with the weighting factor criteria defined in Section 4.3.

3.4.2. External entities

The exchange of eIoCs is performed through MISP, which automatically converts all the received information into the MISP JSON format and stores it in the MISP relational database. The JSON format is always used whenever two or more MISP instances are exchanging intelligence among them. However, when sharing with external entities that do not use MISP, as well as systems which are not able to directly handle the MISP format, the usage of other standards is preferable, also for describing a wider set of TI. From this point of view, STIX 2.0 represents a good choice, being the most used in TI domain [44]. MISP comes out with the possibility of exporting internal stored information using this specific standard. Moreover, the modules in charge to perform the conversion are extensible and can be adapted and improved depending on the organization needs, in particular if they need to develop their own custom export module, and add it to MISP.

After these considerations, the idea behind the *ETIP* is to rely on the MISP JSON format to store incoming events, due to the adoption of MISP. This information is then converted into STIX 2.0, if necessary for the analysis, and exported to the Heuristic Component. This last standard will be considered, starting from the heuristic features identification until the evaluation of the Threat Score, which, once computed, will be added to the original cIoC as a custom attribute. To improve the overall quality of the generated eIoCs, additional information associated to the criteria considered in the score evaluation could be used for the enrichment. Finally, the enriched indicator could be exported using both MISP format or STIX 2.0, depending on the receiver's needs.

4. Threat score evaluation

The threat score evaluation is part of the heuristic component that uses a threat score function (detailed in Section 4.1) to compute the relevance of the received data. The process performs an analysis methodology composed of the following steps:

1. **Source Identification:** during this phase we search and identify all possible sources of information. Examples of these sources are: security logs, databases, report data, OSINT data sources, IoCs, etc.
2. **Heuristics Identification:** different features (e.g., heuristics) are identified from the input data. Such features provide relevant information about the infrastructure (e.g., vulnerabilities, events, faults, errors, etc.) useful in the threat analysis and classification process. Examples of heuristics are: CVE, IP source, IP destination, port source, port destination, timestamp, etc.
3. **Threshold Definition:** for each heuristic, minimum and maximum possible values are defined based on characteristics associated with the instance. We checked, for instance, if the input

data contains or not a CVE for the detected threat. A threshold (e.g., 0–5) is assigned to cover all possible results.

4. **Score Computation:** for each possible instance of the identified heuristic, a score value is assigned based on expert knowledge. All individual scores are then aggregated and a final score is computed. The resulting value will indicate the priority and relevance of the security information coming from OSINT data sources and the monitored infrastructure.
5. **Training Period:** a set of preliminary tests need to be performed during a training process to evaluate the performance of the engine. The tests include real data to analyze the score obtained individually (for each heuristic) and globally (for the whole event) which help to analyze false positive and negative rates.
6. **Engine Calibration:** in order to minimize deviations (e.g., reduce number of false positive, false negative) the engine must be calibrated by analyzing the obtained results, adding other heuristics and/or modifying the assigned values to current attributes.
7. **Final Tests:** Once the engine is calibrated, we can repeat previous tests or add new ones in order to evaluate the performance of the tool.

4.1. Threat score function

The heuristics-based threat score is composed of a set of individual scores as a complement of other prediction tools to indicate the priority and relevance of incoming security information received from OSINT and infrastructure data sources. There exists a large number of different aggregation operators (e.g., Arithmetic Mean [45,46], Geometric Mean [46], Harmonic Mean [46], Weighted Mean [47], Ordered Weighted Averaging — OWA [45,48,49], Weighted Ordered Weighted Aggregation — WOWA [50]) that can be used for the computation of the threat score. They differ on the assumptions about the data (data types) and the type of information that we can incorporate in the model.

From the aforementioned aggregation operators, the Weighted Mean (WM) is the selected function to compute the threat score, due to the following advantages: (i) simple and straightforward function; (ii) avoids indeterminate results and/or null values; (iii) can be used if one or more individual scores are zero; and (iv) individual scores are assumed to have different weights depending on the source and the relevance of the information.

The proposed Threat Score (TS) is defined as the sum of all individual heuristic values (X_i) times its corresponding weight factor (P_i). This latter considers multiple criteria (e.g., relevance, accuracy, timeliness, variety). The sum is then affected to the completeness parameter (C_p), as shown in Eq. (1).

$$TS = C_p \times \left(\sum_{i=1}^{t} X_i \times P_i \right) \tag{1}$$

where

C_p = Completeness criterion: $\frac{Non\_Empty\_Features}{Total\_Features}$

X_i = Value assigned to a given heuristic's feature based on the information obtained from the IoC during the evaluation

P_i = Weighting Criteria

The resulting TS ranges from zero to five ($0 \leq TS \leq 5$), the higher the TS value, the more reliable the IoC. Thus, a TS with a value between zero and one ($0 \leq TS \leq 1$) indicates a Very Low level of priority; a TS with a value between one and two ($1 \leq TS \leq 2$) indicates a Low level of priority; a TS with a value between two and three ($2 \leq TS \leq 3$) indicates a Medium level of priority; a TS with a value between three and four ($3 \leq TS \leq 4$) indicates a High level of priority; and a TS with a value between four and five ($4 \leq TS \leq 5$) indicates a Very High level of priority.

4.2. Heuristic features and values

The first part of the (TS) function refers to the value assigned to a given heuristic (X_i) based on the type of information processed during the evaluation.

Regarding heuristics identification, we considered the STIX 2.1 standard, defined as the de-facto standard for describing threat intelligence [51]. By August 2020, the standard defines eighteen STIX Domain Objects (SDOs [16]) to represent each piece of information with specific attributes that are interrelated for a better understanding and more accurate details on the specific event they represent. Examples of SDOs implemented for the proposed solution to represent cyber threat information in our platform are the following:

- *Attack Pattern*: type of tactics, techniques, and/or procedures describing ways threat actors attempt to compromise targets;
- *Identity*: individuals, organizations, or groups, as well as classes of them that could be involved in a security event;
- *Indicator*: contains patterns used to detect suspicious or malicious cyber activity;
- *Malware*: malicious code or software used to compromise the confidentiality, integrity, or availability of a victim data or system;
- *Tool*: legitimate software that can be used by threat actors to perform attacks;
- *Vulnerability*: mistakes in software that can be directly used by a hacker to gain access to a system or network.

It is important to know that the analysis is not limited only to the aforementioned SDOs, as the solution is designed to be enriched by other objects (e.g., Infrastructure, Location, Report, Threat Actor, etc.), with useful information about potential threats over the target system. For each heuristic, we identified a set of features that indicate valuable information on the identification of a threat. Such features represent Required Common Properties (RCP), Optional Common Properties (OCP), Not Applicable Common Properties (NCP), and Object Specific Properties (OSP) of each STIX Domain Object, as well as infrastructure data that can be useful on the threat identification. Examples of features representing Object Specific Properties are provided in Table 1 and can be obtained from the properties tables of the STIX documentation.[1]

Considering, for instance, that the heuristic to be evaluated is the one corresponding to vulnerabilities, Table 2 summarizes all possible features, attributes, and scores that could be obtained from an IoC of type vulnerability. Please note that besides the required and optional properties associated with the IoC, our approach check other features related to the infrastructure data. For instance, operating_system.

Features related to the vulnerability object have been identified in Table 2, and attributes for each feature type have been associated with a predefined score. This latter is statically assigned based on expert knowledge and an analysis performed on multiple IoCs to determine the possible values included in each attribute. For instance, the operating_system feature presents the following attributes Windows, iOS/Linux, Others, and unknown (if no information is present).

Being Windows the operating system with the highest market share per device type (i.e., Desktop/laptop), with more than 70% of the global market share,[2] and thus the higher attack surface, it is assigned a score of five (5), iOS and Linux, whose attack surface is lower, are assigned a score of three (3), and for other operating systems a score of two (2) has been assigned, leaving a score of one (1) for unknown OSes.

Considering that one of the features to be evaluated is the presence of a Common Vulnerability Exposure (CVE) [43] identified in the input

[1] https://docs.oasis-open.org/cti/stix/v2.1/cs01/stix-v2.1-cs01.html#_nrhq5e9nylke.

[2] https://gs.statcounter.com/os-market-share/desktop/worldwide.

Table 1
Example of heuristic's features.

Heuristics	Object specific properties
Attack pattern	name, description, aliases, kill_chain_phases
Campaign	name, description, aliases, first_seen, last_seen, objective
Course of Action	name, description, action
Grouping	name, description, context, object_refs
Identity	name, description, roles, identity_class, sectors, contact_information
Indicator	name, description, indicator_types, pattern, valid_from, valid_until, kill_chain_phases
Infrastructure	name, description, infrastructure_types, aliases, kill_chain_phases, first_seen, last_seen
Intrusion Set	name, description, aliases, first_seen, last_seen, goals, resource_level, primary_motivation, secondary_motivations
Location	name, description, latitude, longitude, precision, region, country, administrative_area, city, street_address, postal_code
Malware	name, description, malware_types, is_family, aliases, kill_chain_phases, first_seen, last_seen, operating_system_refs, architecture_execution_envs, implementation_languages, capabilities, sample_refs
Malware Analysis	product, version, host_vm_ref, operating_system_ref, installed_software_ref, configuration_version, module, analysis_engine_version, analysis_definition_version, submitted, analysis_started, analysis_ended, result_name, result, analysis_sco_refs, sample_ref
Note	abstract, content, authors, object_refs
Observed Data	first_observed, last_observed, number_observed, objects, object_refs
Opinion	explanation, authors, opinion, object_refs
Report	name, description, report_types, published, object_refs
Threat Actor	name, description, threat_actor_types, aliases, first_seen, last_seen, roles, goals, sophistication, resource_level, primary_motivation, secondary_motivations, personal_motivations
Tool	name, description, tool_types, aliases, kill_chain_phases, tool_version
Vulnerability	name, description

Table 2
Features, attributes and scores associated to an IoC of type vulnerability.

Feature	Description	Attributes and scores
operating_ system	information about the affected operating system	windows (5), iOS, Linux (3), others (2), unknown (1).
source_diversity	IoC has been previously reported by OSINT or different sources	OSINT_source (1); No_OSINT_source (2); infrastructure_source (3).
application	Affected application identified in the IoC	browser (5), office (4), android (3), web (2), other (1).
vuln_app_ in_alarm	Check if incidents/alarms are related to specific applications	present(5), not_present (1).
modified/ created	Timestamp related to object creation/last modification	last_24h (5), last_week (4), last_month (3), last_year (2), other (1).
valid_from	From when the IoC can be considered valid	last_week (4), last_month (3), last_year (2), other (1).
valid_until	Until when the IoC can be considered valid	less_or_equal_to_current_date (1); greater_than_current_date (5).
external_references	External references checked against a local inventory	multi_known_ref (5); single_known_ref (4); unknown_ref (3); no_ref (1).
cve	Check if CVE is found in the information provided by the IoC, and if so, check the CVSS	No CVE or CVE with no CVSS (1), CVE with low CVSS (2), CVE with medium CVSS (3), CVE with high CVSS (4), CVE with critical CVSS (5).

Table 3
Common Vulnerability Score System (CVSS) v3 ratings.
Source: https://www.first.org/cvss/specification-document.

Severity	None	Low	Med	High	Critical
Lower Bound	0.0	0.1	4.0	7.0	9.0
Upper Bound	0.0	3.9	6.9	8.9	10.0

data, the engine will check if the word 'CVE' appears in the input data in order to retrieve the complete CVE number (i.e., CVE-AAAA-NNNN).

If a CVE is found, the engine checks for its associated Common Vulnerability Scoring System (CVSS) [52]. More specifically, the engine will search for its associated base score, which considers access vector, access complexity, authentication, and impact related information based on availability, confidentiality and integrity. Depending on the CVSS score, the vulnerability is labeled as none, low, medium, high or critical, as shown in Table 3.

Each evaluated feature is assigned an individual score based on the defined threshold (e.g., from 1 to 5) that will indicate the level of impact of the feature with respect to the event. We define the variable "Score_CVE" that will compute the individual score value assigned to the presence of a CVE in the input data based on the conditions described in Table 4.

Other features (e.g., source/Destination IP, creation and validity timestamps, etc.) may use higher or lower values in the assignment process. Such individual values are then tuned in the training and calibration processes so that the final threat score reduces the number of false positives and negatives.

4.3. Weighting criteria

The second part of the (TS) function corresponds to the weighting criteria (P_i). According to Henry Dalziel [53], Threat Intelligence refers to specific information that must meet three specific criteria: (i) it must be relevant, for the entity who receives it, (ii) actionable and (iii) valuable, from a business perspective. In [54] the concept of "actionable information" is defined by the European Union Agency for Network and Information Security (ENISA), from an organization point of view as the information that can be used immediately for specific and strategical decision making. Considering [31] and [54], in order to be "actionable", information must meet the following criteria:

Relevance:. It must have some impacts on specific receiver's assets, such as networks, software and hardware. That is, indicators of compromise will usually be considered relevant when a threat could affect the monitored infrastructure. In order to determine the relevance, it is crucial to determine types of threats targeting your assets/systems, considering real-time information (e.g., IoC), from many internal sources, because they are able to provide dynamic and continuous information about current internal monitoring operation, together with a global view of the infrastructure status.

In our analysis, this criterion evaluates if the information associated to a given attribute is useful to identify a threat. Relevance is computed as shown in Table 5.

Timeliness:. Threat intelligence is more reliable when it allows detecting attacker's activity, especially during the same intrusion, to monitor how it evolves during time. Moreover, information about events older than a few hours are, most of the times, irrelevant and non-actionable due to the dynamic nature of some threat's characteristics, considering

Table 4
Examples of individual threat score.

Criteria	Condition	Score
No CVE, or CVE exists with CVSS ‘none’ or 0.0	If CVE ≠ ” & CVSS = ‘none’ | CVSS = 0.0	1
CVE exists with CVSS ‘low’ or less than 4.0	If CVE ≠ ” & CVSS = ‘low’ | CVSS ≤ 4.0	2
CVE exists with CVSS ‘medium’ or less than 7.0	If CVE ≠ ” & CVSS = ‘med’ | CVSS ≤ 7.0	3
CVE exists with CVSS ‘high’ or less than 9.0	If CVE ≠ ” & CVSS = ‘high’ | CVSS ≤ 9.0	4
CVE exists with CVSS ‘critical’ or less than 10.0	If CVE ≠ ” & CVSS = ‘critical’ | CVSS ≤ 10.0	5

Table 5
Weighting criteria values.

Relevance	Score	Timeliness	Score	Accuracy	Score	Variety	Score
No data	1	No data	1	No data	1	No data	1
Attribute has some data with no match	2	Attribute has never been seen	2	Optional Attribute	2	Data come from one source	2
Attribute does not identify threat but helps in the analysis	3	Attribute has been seen with the same value	3	There is a match of one source and the infrastructure	3	Data come from two sources	3
Attribute is useful to identify threat	4	Attribute has been seen with a different value once	4	There is a match of two sources and the infrastructure	4	Data come from more than two sources	4
Mandatory attribute to identify threat	5	Attribute has been seen with a different value more than once	5	There is a match of more than two sources and the infrastructure	5	Data come from all sources	5

that some of them are discovered and analyzed months after the initial compromise.

In our analysis, this criterion evaluates if a detected event is related to an already detected one, by the infrastructure or by the OSINT-based components, and if for instance, such events refer to the same threat, but with a different level of intrusion, providing new or updated information. Timeliness is computed as shown in Table 5.

Accuracy:. The receiver side should be able to process the received data as soon as possible. It depends mainly on three factors, which are the confident of the source from which data is retrieved, the trust level placed in those sources (which, in turn, could depend on factors such as false positives/false negatives rates) and the local dynamic context of the receiver. The latter is crucial in order to avoid inaccurate results and efforts when dealing with incident response.

In our analysis, information coming from OSINT-based components will be compared to the information coming from the infrastructure, if there is a match of one or more attributes, a score will be computed. Accuracy is computed as shown in Table 5.

Variety:. Detection and prevention should not rely on a single technique or tool. It is crucial to use a combination of systems, tools (e.g., IDS, IPS and Firewalls) and sources (e.g., OSINT), especially when they are able to detect the threat at different levels of intrusions (kill chain phases).

In our analysis, this criterion evaluates the sources from where the information is originated or detected e.g., infrastructure, OSINT-based components. Variety is computed as shown in Table 5.

Ingestibility:. Received information must be easy to ingest into internal data management systems for further processing and analysis phases. This is achievable using specific standards for representing this data, allowing the receiver to process data as fast as possible, helping also security analysts, as well as through the usage of specific transfer protocols for sharing the related intelligence.

Ingestibility is not considered in our analysis since we are assuming that all received data is expressed in a structured way and uses a specific standard format to be processed in the system. The data collection will be handled directly by the MISP instance. This criterion would have been meaningful in case of reception of unstructured information, but this scenario is not considered by the threat intelligence sharing platform. The analysis will focus on other criteria with the possibility of adding new ones in the future.

Completeness:. Threat intelligence should provide valuable and complete information to the final receiver, evaluated from the local cyber context point of view of the latter. Sometimes, sources are incomplete when considered alone, but their provided data become actionable once combined or processed with other internal data available to the destination or received from other external sources.

In our analysis, this criterion is used as an overall assessment of the heuristic and not for individual score evaluation of the attributes. Each heuristic is composed of one or more attributes (e.g., CVE is composed of six attributes: (i) no_cve, (ii) cvss_none, (iii) cvss_low, (iv) cvss_medium, (v) cvss_high, (vi) cvss_critical. Completeness is measured as the number of attributes with a non-empty value over the total number of attributes, as shown in Eq. (2).

$$C_p = \frac{Non\_Empty\_Attributes}{Total\_Attributes} \tag{2}$$

In order to perform the heuristic analysis, a value (X_i) must be assigned to each feature (e.g., from 1 to 5). These values correspond to the detected attributes and scores from Table 2 and are based on expert knowledge. They correspond to the usefulness of the criteria in identifying possible threats, malfunctions or anomalies in the monitored infrastructure.

In addition, each feature is affected by a weighting factor (P_i) composed of four criteria: Relevance (R), Accuracy (A), Timeliness (T), and Variety (V). The weighing factor criteria is assessed based on expert knowledge, which determines the P_i value as the total number of points associated with a given feature over the total number of points of all features. Section 5 presents a concrete example of the calculation of these values.

Please note that the proposed threat score is based on Eqs. (1) and (2), which are defined considering a set of heuristics features taken from the information received from the STIX Domain Objects and those contrasted with the infrastructure. Information provided in Tables 1–4 are taking from the literature and re-engineered to be adapted to the described methodology. Attributes and scores are assigned based on expert knowledge and statistical analysis. In addition, although the weighting factor criteria follows the approach of actionable information proposed by ENISA, the categories and scores listed in Table 5 have been carefully defined for the implementation of our component and can be claimed as one of our main contributions. It is worth noting that our approach can be extended with other heuristics, new attributes and additional criteria in the weighting factor evaluation that could incorporate meaningful values to compute the threat score from the data received about threats and malicious events affecting the monitored infrastructure.

Table 6
Infrastructure Inventory.

Nodes	Names	Applications
Node 1	OwnCloud Server	ubuntu, owncloud, ossec, snort, suricata, nids, hids
Node 2	GitLab Server	ubuntu, gitlab, ossec, snort, suricata, nids, hids
Node 3	XL-SIEM Agent	ubuntu, snort, suricata, nids, php
Node 4	XL-SIEM Server	debian, apache, apache storm, apache zookeeper, apache struts, mysql, nessus, openvas
All Nodes	–	linux

5. Case study: MySQL server vulnerability

We have defined an inventory of the infrastructure's network with nodes and the applications already installed. Every eIoC is checked against this information and, if there is a match, the rIoC is generated, associated with a specific node, and, finally, sent to the Output Module. If there is no match, the rIoC is not generated, while, if the match is with a common keyword (e.g., Linux), the new rIoC is associated with all nodes. Table 6 summarizes this information.

Please note that before using *ETIP*, we need to tune the tool using samples of data from the monitored infrastructure. At this point, it is possible to update the values of some attributes composing the threat score evaluation. For instance, instead of assigning a value of 3 to CVEs with medium impact base score, security administrators may consider to assign a value of 2 or 4. End-users are responsible of calibrating the tool according to their infrastructure and data to evaluate. If no calibration is done, the tool will perform the analysis based on the pre-defined values assigned to the corresponding heuristics attributes.

5.1. Input data

An Indicator of Compromise associated with a specific vulnerability was received: CVE-2019-2834: Vulnerability in the MySQL Server component of Oracle MySQL. The severity of this vulnerability is assessed as medium, with CVSS[3] v3.0 equals to 6.5.

The input module receives this IoC, processes it, and sends the resulting cIoC to the *Data Correlator & Exporter* (belonging to MISP) of the Operational Module through a set of API provided by the MISP instance. A specific open source library, written in Python, called PyMISP,[4] exists to interact directly with the MISP platform.

An event coming from the Infrastructure Data Collector is simply stored internally and used later during the heuristic analysis for the threat score evaluation. The other events, instead, which come from the OSINT Data Collector, trigger a built-in automated, and real-time, sharing mechanism, based on the asynchronous messaging library zeromq,[5]allowing the Heuristic Component to receive them, in STIX 2.0 format, and start the correlation with the stored infrastructure data. Once the threat score is computed, the eIoC is generated enriching the MISP JSON version of the cIoC, stored in the MISP database, adding the threat score as a new MISP attribute.

Therefore, the heuristic component receives the cIoC regarding the MySQL vulnerability, combines it with the data it receives from the infrastructure's network and evaluates the result, obtaining both eIoC and threat score.

Table 7
Threat Score Results.

Feature	X_i	R	A	T	V	Total	P_i
operating_system	3	4	5	5	3	17	0.1429
source_diversity	3	3	4	5	4	16	0.1345
application	5	4	3	4	4	15	0.1261
vuln_app_in_alarm	5	2	2	3	3	10	0.8040
modified/created	3	3	2	2	2	9	0.0756
valid_from	3	3	2	3	1	9	0.0756
valid_until	5	3	2	4	1	10	0.0840
external_references	4	4	5	4	3	16	0.1343
CVE	3	5	4	5	3	17	0.1429

5.2. Preliminary results

By contrasting the information of cIoC with the list of features presented in Table 2, we identified that the reported security incident: - affects the Debian OS and the MySQL; - was first reported from OSINT; - there are no alerts from the monitored infrastructure related to MySQL application; - was created and last modified on 2019-09-13; - is valid for one year; - there exists external references from the Common Attack Pattern Enumeration and Classification (CAPEC[6]) and the Common Vulnerabilities and Exposures (CVE[7]).

Table 7 summarizes the assessment results associated with the cIoC from a remote code execution. Each feature is assigned a heuristic value, for instance, according to the description of the incident, the affected operating system is Debian, therefore $X_i = 3$ for this feature, the CVE has a medium CVSS, therefore $X_i = 3$ for this feature.

The value of P_i is computed after the RATV assessment (following the guidelines from Table 5). The total RATV value for this example is 119, from which the feature about operating system has computed 17 points, therefore $P_i = 17/119 = 0.1429$. The process is repeated for every feature.

Following Eq. (1), and using the scores values presented in Table 7, we compute the threat score for the MySQL vulnerability as: $TS_{(MySQL)} = \frac{9}{9} \times \left(\sum_{i=1}^{t} X_i \times P_i\right) = 3.72$ (more examples of Threat Score computation and details on the use case analysis can be found at https://caisplatform.wixsite.com/english).

5.3. Information sharing and visualization

Fig. 6 depicts the visualization data related to the affected node in our proposed platform. Please note that the visualization data presented by *ETIP* are related to the network topology, the assets that belong to the infrastructure, and the security data related to each asset (e.g., alarms, vulnerabilities, threat score data).

The left part of Fig. 6 shows the affected node (i.e., XL-SIEM SERVER) with two associated alarms, both with low severity (green) and four reduced IoCs; as well as more details about the node, such as node type, IP address, operating system. The right part of Fig. 6 provides the information contained in alarms (green) and rIoCs (blue). These detailed information is obtaining by clicking on top of the node graphical representation.

As shown in Fig. 6 a reduced IoC (rIoC) is generated by *ETIP* regarding the CVE-2019-2834 vulnerability the platform received, which contains the information we presented above (i.e., CVE, description, and the affected infrastructure). A Threat Score of 3.72 points indicates the associated severity of the vulnerability with the identified CVE ID, which helps security analysts in their prioritization of security actions.

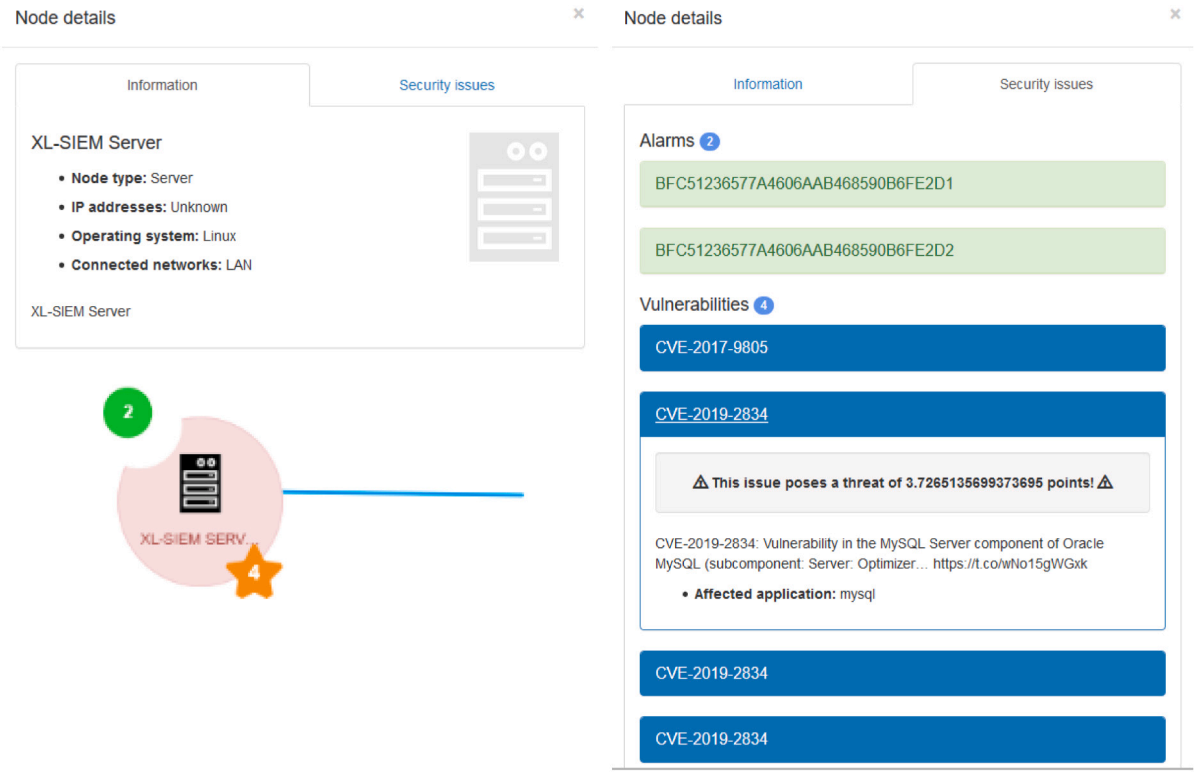
[3] https://nvd.nist.gov/vuln/detail/CVE-2019-2834.

[4] https://github.com/MISP/PyMISP.

[5] http://zeromq.org/.

[6] https://capec.mitre.org/.

[7] https://cve.mitre.org/.



Node details ×

Information | Security issues

XL-SIEM Server

- **Node type:** Server
- **IP addresses:** Unknown
- **Operating system:** Linux
- **Connected networks:** LAN

XL-SIEM Server

Node details ×

Information | Security issues

Alarms 2

BFC51236577A4606AAB468590B6FE2D1

BFC51236577A4606AAB468590B6FE2D2

Vulnerabilities 4

CVE-2017-9805

CVE-2019-2834

⚠ **This issue poses a threat of 3.7265135699373695 points!** ⚠

CVE-2019-2834: Vulnerability in the MySQL Server component of Oracle MySQL (subcomponent: Server: Optimizer… https://t.co/wNo15gWGxk

- **Affected application:** mysql

CVE-2019-2834

CVE-2019-2834

Fig. 6. Security visualization data.

6. Evaluation of *ETIP* components and analysis of results

In this section we will illustrate the advantages of our approach, and the benefits our platform will bring in terms of prioritizing threat information received from external sources (e.g., OSINT). We will use the Threat Intelligence Sharing Module to evaluate relevance, accuracy, and other features on the information received from the OSINT Data Collector Module in the form of composed IoCs. For the threat score evaluation we will consider both: the data coming from the infrastructure through security systems and tools (e.g., SIEMs, IDS), as well as the cIoCs obtained by open source and public feeds. The resulting threat score will be inserted in the information associated with the analyzed cIoC. The higher the threat score value, the higher the priority of the associated information when handled by incident response teams and security analysts.

The OSINT Data Collector module will provide both: single and composed IoCs to the Threat Intelligence Sharing module. We expect that the composed IoCs will have an associated threat score higher than the threat score of each single IoCs they contain. The entire process considered in this use case is characterized by four sequential phases: **Collecting** and **Aggregating** phases, performed by the OSINT Data Collector module, followed by the **Sharing** phase, which involves both modules, and the **TS Evaluation** phase, completely handled by the Threat Intelligence Sharing module.

6.1. *Collecting phase:*

In order to collect OSINT data we configured a MISP instance with 34 OSINT feeds from higher value information (e.g., CVE vulnerabilities) and low value information (e.g., IP blacklists). These feeds are provided by diverse public free entities and reach MISP in different formats, such as csv and txt files. OSINT data are normalized in a single format, namely the MISP format, and then stored as IoCs in the MISP database. Afterwards, the Deduplicator module we developed is executed to load the IoCs and search for duplicates to delete them. This task allows improving MISP in two forms: identify duplicated IoCs and reduce the quantity of data stored, and therefore, increasing the MISP performance.

6.2. *Aggregating phase:*

After removing the duplicated IoCs, the IoC Aggregator component analyzes the resulting IoCs to look for connections among them. For the connections found, the aggregator puts the involved IoCs in the same group of IoCs, since they are related to the same malicious threat. At the end, we have several and different groups of IoCs forming clusters, each one for a particular threat category. At the point of view of a threat category, a cluster can contain IoCs correlated between them and related with a same (sub-)threat (or attack) and possibly with other valuable malicious information that can be provided in a same IoC. This means that a cluster can contain sub-clusters of IoCs regarding to different attacks. Such sub-clusters can well characterize, this point of view, attacks that have been executed, for which individual IoCs could not allow their identification. Finally, each sub-cluster is represented as one IoC, i.e., all its IoCs are merged in a single one, generating a composed IoC, and then, they are stored in the MISP database.

6.3. *Sharing phase:*

The final outcome of the OSINT Data Collector module is sent to the threat intelligence sharing module for proceeding with the computation of the threat score. This integration is achieved easily thanks to the adoption of MISP.

More precisely, two different MISP instances are used, one by each module. For simplicity, and for facilitating reader comprehension, we will refer to them as $MISP_A$ and $MISP_B$, respectively. These instances have been synchronized between each other to allow a real-time and completely automated information sharing, following the guidelines provided in the MISP book [38] for setting up a MISP synchronization server on $MISP_A$. This server has been associated with a specific user with synchronization privileges, which is replicated in both instances. Injecting, and publishing IoCs in $MISP_A$ on behalf of this user, triggers a push operation of one or more IoCs directly on $MISP_B$, completing the one-way information sharing needed for this use case scenario.

When the synchronization server is set up, the sync user authentication key must be specified. This information is provided by $MISP_B$, when the user is created.

6.4. TS evaluation phase:

In order to perform the computation of the threat score, $MISP_B$ needs to be extended with new functionalities. For this specific use case, we developed a new MISP export module, integrating it with the core software of the platform, following the guidelines [55] provided by the MISP community and developers. In this way, the module is available directly from the MISP UI, where it can be triggered manually by the user for a specific IoC, to retrieve and send the IoC directly to the Heuristic Module of the platform (Fig. 4), where the threat score function has been implemented, and added to the original IoC as a new MISP attribute.

Aiming at correlating cIoCs with infrastructure data, as well as with other useful information about cyber events from open source and public feeds, a MongoDB [42] database is used, and the information is stored as JSON documents. The final IoC (i.e., enriched IoC) could be shared with specific security tools or internal SOCs and CSIRTs, with the additional threat score used for determining the priority of the contained data, in case of some defense activities would be needed.

To provide practical examples of the functionality of our platform, eleven samples of composed IoCs were considered, and the entire process previously described was executed in each of them. As a result, the threat score (TS) is computed for every single IoC ($sIoC$) integrating the composed ones, making it possible to compare with the threat score computed for the composed IoCs ($cIoC$).

The internal dataset used for the correlation is composed by events detected by the proprietary Cross-Layer SIEM (XL-SIEM) [56], and a set of blacklisted IP addresses, malicious URLs and domain names, retrieved from some MISP Open Source feeds [23].

For the heuristic analysis, a subset of nine MISP attributes [38] was selected, composed of the ones which are more relevant according to the monitored infrastructure (i.e., vulnerability, filename, src-IP, dst-IP, hostname, domain, url, link, and md5). This does not mean that other attributes are discarded, they simply have a higher importance when specific criteria are evaluated, especially for relevance and completeness.

Results are summarized in Table 8 when we evaluate these IoCs with *ETIP*.[8] Each row of the table is associated with a sample of composed IoCs (e.g., $S_1,\dots,S_{11}$), specifying the number of single IoCs ($sIoC$) composing them, their individual Threat Score (TS), and the global TS of the composed IoC ($cIoC$).

As depicted in Table 8, in most of the cases, the TS of the composed IoCs is higher than the TS for each of the related single ones. This improvement is strictly dependent on two main factors:

1. The number of attributes present in the IoC. The higher this number, the higher the probability of increasing the overall quality when the aggregation is performed; and
2. The quality of the single IoCs. The higher the quality of the information found in the attributes present in the sIoCs, the lower the probability to increase the overall quality when aggregating several IoCs.

For the first factor, cIoCs S_6 and S_9 have a high number of single IoCs (17 and 11 IoCs respectively), for which the global TS of them has greatly improved compared to the one of each $sIoC$. More precisely, the highest sIoC threat score in S_6 is 2.66, whereas the corresponding cIoC threat score is 3.98.

For the second factor, we have cIoCs S_2, S_3, S_7, and S_{10}, in which the aggregation process is not able to add a relevant level of quality to the final IoC, the cIoC. In these cases, the quality of the information identified in the $sIoC$ samples results in high TS values. Although in most of the cases, the TS value of the $cIoC$ is higher than the one associated to each $sIoC$, the improvement is low, having in some cases a lower TS value in the $cIoC$ compared to one of the $sIoC$ (i.e., S_7).

It is important to note that among all the criteria used in the TS computation, the completeness (i.e., C_p) is the criterion that affects the most the final result. Whereas, all other criteria are adding individual values to the TS, the completeness criterion is multiplying to the overall addition, affecting to a higher level the TS results of single or composed IoCs.

However, this is not always true. Indeed, the only example where the Threat Score of the composed IoC is lower than one or more single IoCs (two in this specific case) is the one where 7 single IoCs are considered. This is caused by the second e factors mentioned above. When some single IoCs have already a good quality with respect to the others, it happens that the aggregation process, which does not consider any known information about the monitored infrastructure, instead of improving the overall quality of the final IoC, fills it with less relevant information than the one inherited from the “good” single IoCs, obtaining the decrease of the Threat Score. These specific subcases are not predictable, because we do not actually know what kind of single IoCs the Operational module will receive from the OSINT Data Collector component, which is not aware of the peculiarities of the infrastructure monitored by the former.

In a similar way, when few single IoCs are considered, the aggregation process is not able to add a relevant level of quality to the final IoC, as can be checked consulting the cases with 2 or 3 single IoCs. They are not characterized by huge differences in terms of Threat Scores, indeed the Threat Score of the final IoC will still be higher, but, at the same time, very close to ones evaluated for the singles.

7. Conclusions and perspective for future work

This paper presents *ETIP*, an enriching threat intelligence platform, as an extended import, quality assessment processes and information sharing capabilities in current Threat Intelligence Platforms (TIPs). The proposed platform gathers and processes structured information from external sources (e.g., OSINT sources) and from the monitored organization’s network infrastructure. *ETIP* is composed of three main modules: (i) a Input Module, in charge of collecting, normalizing, processing and aggregating indicators of compromise (IoCs) from OSINT feeds, and collecting static and real time information from the monitored infrastructure; (ii) an Operational Module, able to correlate both collected data to generate enriched IoCs, and assess this new data by scoring the threat information it comprises; and (iii) an Output Module, aiming at presenting results graphically and sharing them among external entities to improve the prevention and detection capabilities of defense mechanisms (e.g., SiEMs, IDS) against cybercrime.

The *ETIP* platform computes a Threat Score (TS) associated with each IoC before sharing it with both internal monitoring systems and tools and trusted external parties. Enriched IoCs will contain a threat score that will enable SOC analysts to prioritize the analysis of security incidents. The TS evaluates heuristics with two types of weights: (i) individual weights assigned to every attribute (e.g., relevance, accuracy, variety, etc.); and (ii) global weight (i.e., completeness criterion) assigned to the heuristic. The higher the TS value, the more reliable the IoC. Thus, as the TS value approaches to zero, the IoC can be

[8] More information about *ETIP* platform can be found in https://caisplatform.wixsite.com/english.

Table 8
Threat Score results of composed IoCs and their individual IoCs.

Samples	N. of sIoCs	Individual TS	Global TS
S_1	5	1.86, 2,55, 1.80, 0.71, 1.94	3.18
S_2	3	1.43, 2.32, 1,58	2.53
S_3	3	2.48, 1.54, 1.09	2.87
S_4	6	1.18, 1.40, 1.54, 0.64, 1.41, 2.03	3.07
S_5	2	2.84, 1.66	3.22
S_6	17	1.39, 2.22, 2.21, 1.99, 1.87, 0.70, 1.66, 1.10, 0.56, 0.96, 0.94, 0.56, 1.58, 2.66, 2.27, 1.36, 1.08	3.98
S_7	7	2.09, 3.27, 1.89, 0.89, 2.88, 1.93, 1.66	2.84
S_8	4	3.06, 2.68, 2.11, 1.55	3.11
S_9	11	1.66, 1.21, 2.35, 1.92, 1.33, 1.29, 1.6, 0.90, 0.88, 1.02, 0.56	4.13
S_{10}	2	2.43, 2.31	2.54
S_{11}	2	0.99, 0.55	1.29

considered as poor, incomplete and/or not reliable with a very low priority level.

The paper also presents an evaluation of *ETIP* in a real use-case scenario and an assessment of its components, by evaluating them with aggregated IoCs and single IoCs that compose the aggregated ones. In both evaluations we verified that the former is valuable to represent threats since they relate information of different IoCs regarding the same threat, complementing thus each other. Also, some single IoCs, when analyzed individually, do not add value in the detection of threats since they do not carry relevant information about a given threat, but when aggregated with other IoCs they contribute to complete the information about the threat. In addition, the TS of aggregated IoCs is higher than their single IoCs, denoting that the heuristics we proposed to evaluate the threat score are reliable and capable of measuring threats. Finally, thanks to the aggregation of IoCs and their correlation with the organization's infrastructure data, it was possible to detect a vulnerability in the MySQL DBMS used in the real use-case scenario, which was not possible by only analyzing both types of data separately.

ETIP has been tested and validated in production and operational environments using three industrial SIEM solutions: Micro Focus ArcSigh,[9] SIEMonster,[10] and the XL-SIEM [56], for which, the generated output has been connected with the Malware Information Sharing Platform (MISP). Results have shown a promising and interesting approach to assess the relevance of data coming from internal components (e.g., infrastructure data) and external components (e.g., OSINT data) in the detection of threats and malicious network activities. Future work will focus on performing additional testing and validation of the components, heuristics and attributes used in the analysis of IoCs. In addition, we plan to develop new attributes to enrich the threat score analysis, improving the quality of the refined threat intelligence to be shared, providing not only the final threat score, but also detailed information about each single criterion used in the evaluation, which in turn helps to improve threat detection and incident response. Furthermore, it is important to use functions based on precision and recall parameters, to evaluate the performance of the proposed solution and compare up to which extent false rates are minimized.

CRediT authorship contribution statement

Gustavo González-Granadillo: Conceptualization, Methodology, Software, Validation, Formal analysis, Supervision, Project administration, Investigation, Writing - original draft, Visualization. **Mario Faiella:** Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing - original draft, Visualization. **Ibéria Medeiros:** Conceptualization, Methodology, Software, Validation, Formal analysis, Supervision, Project administration, Funding acquisition, Investigation, Writing - original draft, Visualization. **Rui Azevedo:** Conceptualization, Methodology, Validation, Formal analysis, Software, Investigation, Visualization. **Susana González-Zarzosa:** Conceptualization, Methodology, Validation, Formal analysis, Investigation, Supervision, Visualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by the EC through the DiSIEM project (H2020-700692) and the NeCS project (H2020-675320), and by the FCT, Portugal through the SEAL project (PTDC/CCI-INF/29058/2017) and the LASIGE Research Unit, Portugal (UIDB/00408/2020).

References

[1] CEA. The cost of malicious cyber activity to the U.S. economy. 2018, Online, available at https://www.whitehouse.gov/wp-content/uploads/2018/03/The-Cost-of-Malicious-Cyber-Activity-to-the-U.S.-Economy.pdf.

[2] Ventures C. 2017 cybercrime report. 2017, Online, available at https://cybersecurityventures.com/cybercrime-damages-6-trillion-by-2021/.

[3] Accenture. Cost of Cyber Crime Study. Insights of the security investments that make a difference. 2017, Online, available at https://www.accenture.com/t20170926t072837z__w__/us-en/_acnmedia/pdf-61/accenture-2017-costcybercrimestudy.pdf.

[4] Liao X, Yuan K, Wang X, Li Z, Xing L, Beyah. R. Acing the IOC Game: Toward automatic discovery and analysis of open-source cyber threat intelligence. In: ACM SIGSAC conference on computer and communications security; 2016. p. 755–66.

[5] Sabottke C, Suciu O, Dumitras T. Vulnerability disclosure in the age of social media: exploiting twitter for predicting real-world exploits. In: 24th USENIX security symposium; 2015. p. 1041–56.

[6] Campiolo R, Santos L, Batista DM, Gerosa MA. Evaluating the utilization of Twitter messages as a source of security alerts. In: 28th annual ACM symposium on applied computing; 2013.

[7] Alves F, Bettini A, Ferreira PM, Bessani A. Processing tweets for cybersecurity threat awareness. Technical report, 2019, available at https://arxiv.org/pdf/1904.02072.pdf.

[8] Alves F, Ferreira PM, Bessani A. Design of a classification model for a Twitter-based streaming threat monitor. In: International conference on dependable systems and networks workshops. IEEE; 2019.

[9] Sillaber C, Sauerwein C, Mussmann A, Breu R. Data quality challenges and future research directions in threat intelligence sharing practice. In: ACM on workshop on information sharing and collaborative security. ACM; 2016, p. 65–70.

[10] Mavroeidis V, Bromander S. Cyber threat intelligence model: An evaluation of taxonomies, sharing standards, and ontologies within cyber threat intelligence. In: Intelligence and security informatics conference. IEEE; 2017, p. 91–8.

[11] Faiella M, Gonzalez-Granadillo G, Medeiros I, Azevedo R, Gonzalez-Zarzosa S. Enriching threat intelligence platforms. In. Conference on security and cryptography, SECRYPT. Czech Republic; 2019.

[12] Skopik F, Settanni G, Fiedler R. A problem shared is a problem halved: A survey on the dimensions of collective cyber defense through security information sharing. Comput Secur 2016;60:154–76.

[9] https://www.microfocus.com/en-us/products/siem-security-information-event-management/overview.

[10] https://siemonster.com/.

[13] Hernandez-Ardieta J, Tapiador J, Suarez-Tangil G. Information sharing models for cooperative cyber defence. In: 5th International conference on cyber conflict (cycon); 2013. p. 1–28.

[14] Ring T. Threat intelligence: why people don't share. Computer Fraud & Security; 2014, p. 5–9.

[15] Darknet. OpenIOC - Sharing Threat Intelligence. 2016, Online, available at https://www.darknet.org.uk/2016/06/openioc-sharing-threat-intelligence/.

[16] OASIS. Introduction to STIX. 2020, Consulted on June 2020. Online, available at https://oasis-open.github.io/cti-documentation/stix/intro.html.

[17] OASIS. Introduction to TAXII. 2020, Consulted on June 2020. Online, available at https://oasis-open.github.io/cti-documentation/taxii/intro.html.

[18] Scarfone K. Comparing the best SIEM systems on the market. 2015, Online, available at http://searchsecurity.techtarget.com/feature/Comparing-the-best-SIEM-systems-on-the-market.

[19] Kavanagh KM, Rochford O, Bussa T. 2016 magic quadrant for SIEM. Gartner technical report G00290113, 2016.

[20] Sheridan K. Future of the SIEM. 2017, Dark Reading, threat intelligence article.

[21] Caccia R, Cassetto O, Shteiman B. The future of SIEM. 2017, International Information Systems Security Certification Consortium (ISC^2) Webminar available at https://www.brighttalk.com/.

[22] Tounsi W, Rais H. A survey on technical threat intelligence in the age of sophisticated cyber attacks. Comput Secur 2018;72:212–33.

[23] MISP. Open source threat intelligence platform & open standards for threat information sharing. 2019, Online, available at http://www.misp-project.org.

[24] CSIRTG. The fastest way to consume threat intelligence. 2019, Online, available at https://csirtgadgets.com/collective-intelligence-framework.

[25] MITRE. CRITS: Collaborative research into threats. 2019, Online, available at https://crits.github.io/.

[26] Leonard J. TheHive Project: Open source, free and scalable cyber threat intelligence & security incident response solutions. 2019, Online, available at https://blog.thehive-project.org/tag/soltra-edge/.

[27] Sauerwein C, Sillaber C, Mussmann A, Breu R. Threat intelligence sharing platforms: An exploratory study of software vendors and research perspectives. In: Conference on wirtschaftsinformatik; 2017.

[28] ENISA. Exploring the opportunities and limitations of current Threat Intelligence Platforms. ENISA; 2017, Online available at https://www.enisa.europa.eu/publications/exploring-the-opportunities-and-limitations-of-current-threat-intelligence-platforms.

[29] Owen T. Threat intelligence and SIEM. In: Masters research project. Lewis University; 2015.

[30] LogRhythm. Build your SOC on a powerful foundation. LogRhythm NextGen SIEM platform. 2019, Online, available at https://logrhythm.com/.

[31] ThreatConnect. Threat Intelligence Platforms. Everything You've Ever Wanted to Know But Didn't Know to Ask. Ebook; 2018, Accessed February 2018.

[32] Trabelsi S, et al. Mining social networks for software vulnerabilities monitoring. In: 7th International conference on new technologies, mobility and security (NTMS); 2015.

[33] Ritter A, Wright E, Case W, Mitchell T. Weakly supervised extraction of computer security events from twitter. In: Proceedings of the 24th international conference on world wide web; 2015.

[34] Mittal S, Das PK, Mulwad V, Joshi A, Finin T. CyberTwitter: Using Twitter to generate alerts for cybersecurity threats and vulnerabilities. In: International symposium on foundations of open source intelligence and security informatics; 2016.

[35] Mittal S, Joshi A, Finin T. Thinking, fast and slow: Combining vector spaces and knowledge graphs. Technical report, 2017, Retrieved from http://arxiv.org/abs/1708.03310.

[36] Nunes E, Diab A, Gunn A, Marin E, Mishra V, Paliath V, et al. Darknet and deepnet mining for proactive cybersecurity threat intelligence. Technical report, 2016, Retrieved from http://arxiv.org/abs/1607.08583.

[37] Queiroz A, Keegan B, Mtenzi F. Predicting software vulnerability using security discussion in social media. In: 16th European conference on cyber warfare and security. 2017, p. 628–34, Retrieved from https://www.scopus.com/inward/record.uri?eid=2-s2.0-85027995432&partnerID=40&md5=7c9399c8e40b02319c09b9f127ccdcd2.

[38] Sharing MT. MISP User guide - A threat sharing platform. 2019, Online, available at https://www.circl.lu/doc/misp/book.pdf.

[39] Lord N. What are indicators of compromise? 2018, Online, available at https://digitalguardian.com/blog/what-are-indicators-compromise.

[40] Azevedo R, Medeiros I, Bessani A. PURE: Generating quality threat intelligence by clustering and correlating OSINT. In: 18th IEEE international conference on trust, security and privacy in computing and communications, TrustCom; 2019. p. 483–90.

[41] ZeroMQ. An open-source universal messaging library. 2019, Online, available at http://zeromq.org/.

[42] MongoDB. The database for modern applications. 2019, Online, available at https://www.mongodb.com/.

[43] CVE. Common vulnerabilities and exposures. 2019, Online, available at https://cve.mitre.org/.

[44] Tounsi W, Rais H. A survey on technical threat intelligence in the age of sophisticated cyber attacks. Comput Secur 2018;72:212–33.

[45] Torra V, Narukawa Y. Modeling decisions: Information fusion and aggregation operators. Springer-Verlag Berlin Heidelberg; 2007.

[46] Ravana S, Moffat A. Score aggregation techniques in retrieval experimentation. In: Twentieth Australasian database conference; 2009.

[47] Torra V. Aggregation functions and information fusion. Modeling decisions. 2017, Online, available at http://www.mdai.cat/ifao/slides/transparencies.SFLA.2017.pdf.

[48] Derakhshandeh S, Mikaeilvand N. Fuzzy method for identification of aggregate weights in ordered weighted averaging operators. Middle-East J Sci Res 2011;7(3):293–5.

[49] Cornelis C, Victor P, Herrera-Viedma E. Ordered Weighted Averaging Approaches for Aggregating Gradual Trust and Distrust. In: XV Spanish congress on technology and fuzzy logic ESTYLF; 2010. p. 555–60.

[50] Damiani E, Vimercati SD, Samarati P, Viviani M. A WOWA-based aggregation technique on trust values connected to metadata. J Electron Notes Theoret Comput Sci 2006;157:131–42.

[51] Sauerwein C, Sillaber C, Mussmann A, Breu R. Threat intelligence sharing platforms: An exploratory study of software vendors and research perspectives. In: International conference on wirtschaftsinformatik; 2017.

[52] FIRST. Common vulnerability scoring system version 3.1: Specification document. 2019, Online, available at https://www.first.org/cvss/specification-document.

[53] Dalziel H. How to define and build an effective cyber threat intelligence capability. In: Syngress, eBook; 2014.

[54] ENISA.

[55] MISP. Modules for expansion services, import and export in MISP. 2019, Online, available at https://github.com/MISP/misp-modules.

[56] Gonzalez-Granadillo G, Gonzalez-Zarzosa S, Faiella M. Towards an enhanced security data analytic platform. In: 15th international conference on security and cryptography (SECRYPT); 2018.