UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

Generating Threat Intelligence based on OSINT and a Cyber Threat Unified Taxonomy

Cláudio Dinis Neves Martins

Mestrado em Segurança Informática

Dissertação orientada por:
Prof. Doutora Ibéria Vitória de Sousa Medeiros

2020

Acknowledgments

I express my sincere thanks to my advisor, Prof Ibéria Medeiros, who always pushed me to think about my research from as many aspects as possible and who was always respectfully concerned with me.

This work was supported by the European Commission through the H2020 programme under grant agreement 700692 (DiSIEM), and LASIGE Research unit, ref.UIDB/00408/2020.

Resumo

As ameaças cibernéticas atuais utilizam múltiplos meios de propagação, tais como a engenharia social, vulnerabilidades de e-mail e aplicações e, muitas vezes, operam em diferentes fases, tais como o comprometimento de um único dispositivo, o movimento lateral na rede e a exfiltração de dados. Estas ameaças são complexas e dependem de táticas bem avançadas, por forma a passarem despercebidas nas defesas de segurança tradicionais, como por exemplo firewalls. Um tipo de ameaças que tem tido um impacto significativo na ascensão do cibercrime são as ameaças persistentes avançadas (APTs), as quais têm objetivos claros, são altamente organizadas, têm acesso a recursos praticamente ilimitados e tendem a realizar ataques ocultos por longos períodos e com múltiplas tentativas. À medida que as organizações têm tido consciência que os ciberataques estão a aumentar em quantidade e complexidade, a utilização de informação sobre ciberameaças está a ganhar popularidade para combater tais ataques. Esta tendência tem acompanhado a evolução das APTs, uma vez que estas exigem um nível de resposta diferente e mais específico a cada organização. A informação sobre ciberameaças pode ser obtida de diversas fontes e em diferentes formatos, sendo a informação de fonte aberta (OSINT) uma das mais comuns. Também pode ser obtida por plataformas especificas de ameaças (TIPs) que ajudam a consumir, produzir e partilhar informações sobre ciberameaças. As TIPs têm múltiplas vantagens que permitem às organizações explorar facilmente os principais processos de recolha, enriquecimento e partilha de informações relacionadas com ameaças. No entanto, devido ao elevado volume de informação OSINT recebido por dia e às diversas taxonomias existentes para classificação de ciberameaças provenientes do OSINT, as TIPs atuais apresentam limitações de processamento desta, capaz de produzir informação inteligente (threat intelligence, TI) de qualidade que seja útil no combate de ciberataques, impedido assim a sua adoção em massa. Por sua vez, os analistas de segurança desperdiçam um tempo considerável em analisar o OSINT e a classificá-lo com diferentes taxonomias, por vezes, correspondentes a ameaças da mesma categoria.

Esta dissertação propõe uma solução, denominada *Automated Event Classification and Correlation Platform* (AECCP), para algumas das limitações das TIPs mencionadas anteriormente e relacionadas com a gestão do conhecimento de ameaças, a triagem de ameaças, o elevado volume de informação partilhada, a qualidade dos dados, as capacidades de análise avançadas e a automatização de tarefas. Esta solução procura aumentar a qualidade da TI produzidas por TIPs, classificando-a em conformidade com um sistema de classificação comum, removendo a informação irrelevante, ou seja, com baixo valor, enriquecendo-a com dados importantes e relevantes de fontes OSINT, e agregando-a em eventos com informação semelhante. O sistema de classificação comum, denominado de *Unified Taxonomy*, foi definido no âmbito desta dissertação e teve como base uma análise de outras taxonomias públicas conhecidas e utilizadas na partilha de TI.

O AECCP é uma plataforma composta por componentes que podem trabalhar em conjunto ou individualmente. O AECCP compreende um classificador (*Classifier*), um redutor de informação irrelevante (*Trimmer*), um enriquecedor de informação baseado em OSINT (*Enricher*) e um agregador de agregador de eventos sobre a mesma ameaça, ou seja, que contêm informação semelhante (*Clusterer*). O *Classifier* analisa eventos e, com base na sua informação, classifica-os na *Unified Taxonomy*, por forma a catalogar eventos ainda não classificados e a eliminar a duplicação de taxonomias com o mesmo significado de eventos previamente classificados. O *Trimmer* elimina a informação menos pertinente dos eventos baseando-se na classificação do mesmo. O *Enricher* enriquece os eventos com dados externos e provenientes de OSINT, os quais poderão conter informação importante e relacionada com a informação já presente no evento, mas não contida no mesmo. Por último, o *Clusterer* agrega eventos que partilham o mesmo contexto associado à classificação de cada um e à informação que estes contêm,

produzindo aglomerados de eventos que serão combinados num único evento. Esta nova informação garantirá aos analistas de segurança o acesso e fácil visibilidade a informação relativa a eventos semelhantes aos que estes analisam.

O desenho da arquitetura do AECCP, foi fundamentado numa realizada sobre três fontes públicas de informação que continham mais de 1100 eventos de ameaças de cibersegurança partilhados por 24 entidades externas e colecradas entre os anos de 2016 e 2019. A *Unified Taxonomy* utilizada pelo *Classifier*, foi produzida com base na análise detalhada das taxonomias utilizadas por estes eventos e nas taxonomias mais utilizadas na comunidade de partilha de TI sobre ciberameaças. No decorrer desta análise foram também identificados os atributos mais pertinentes e relevantes para cada categoria da *Unified Taxonomy*, através da agregação da informação em grupos com contexto semelhante e de uma análise minuciosa da informação contida em cada um dos mais de 1100 eventos.

A dissertação, também, apresenta os algoritmos utilizados na implementação de cada um dos componentes que compõem o AECCP, bem como a avaliação destes e da plataforma. Na avaliação foram utilizadas as mesmas três fontes de OSINT utilizadas na análise inicial, no entanto, com 64 eventos criados e partilhados mais recentemente que os utilizados nessa análise. Dos resultados, foi possível verificar um aumento de 72% na classificação dos eventos, um aumento médio de 54 atributos por evento, com uma redução nos atributos com pouco valor e aumento superior de atributos com maior valor, após os eventos serem processados pelo AECCP. Foi também possível produzir 24 eventos agregados, enriquecidos e classificados pelos outros componentes do AECCP. Por último, foram processados pelo AECCP 6 eventos com grande volume de informação produzidos por uma plataforma externa, denominada de PURE, onde foi possível verificar que o AECCP é capaz de processar eventos oriundos de outras plataformas e de tamanho elevando.

Em suma, a dissertação apresenta quatro contribuições, nomeadamente, um sistema de classificação comum, a *Unified Taxonomy*, os atributos mais pertinentes para cada uma das categorias da *Unified Taxonomy*, o desenho da arquitetura do AECCP composto por 4 módulos (*Classifier*, *Trimmer*, *Enricher* e *Clusterer*) que procura resolver 5 das limitações das atuais TIPs (gestão do conhecimento de ameaças, a triagem de ameaças, o elevado volume de informação partilhada, a qualidade dos dados e as capacidades de análise avançadas e a automatização de tarefas) e a sua implementação e avaliação.

Palavras-chave: cibersegurança, open source intelligence (OSINT) / informação de fonte aberta, indicadores de comprometimento (IoCs), plataformas de partilha de informações sobre ameaças (TIP), informação sobre ameaças (TI) de qualidade, classificação automática de TI

Abstract

Today's threats use multiple means of propagation, such as social engineering, email, and application vulnerabilities, and often operate in different phases, such as single device compromise, network lateral movement and data exfiltration. These complex threats rely on well-advanced tactics for appearing unknown to traditional security defences. One type that had a major impact in the rise of cybercrime are the advanced persistent threats (APTs), which have clear objectives, are highly organized and well-resourced and tend to perform long term stealthy campaigns with repeated attempts. As organizations realize that attacks are increasing in size and complexity, threat intelligence (TI) is growing in popularity and use amongst them. This trend followed the evolution of the APTs as they require a different level of response that is more specific to the organization. TI can be obtained via many formats, being open source intelligence (OSINT) one of the most common; and using threat intelligence platforms (TIPs) that aid organization consuming, producing and sharing TI. TIPs have multiple advantages that enable organisations to easily bootstrap the core processes of collecting, normalising, enriching, correlating, analysing, disseminating and sharing of threat related information. However, current TIPs have some limitations that prevents theirs mass adoption. This dissertation proposes a solution to some of these limitations related *with threat knowledge management*, *limited technology enablement in threat triage*, *high volume of shared threat information*, *data quality* and *limited advanced analytics capabilities and tasks automation*. Overall, our solution improves the quality of TI by classifying it accordingly a common taxonomy, removing the information with low value, enriching it with valuable information from OSINT sources, and aggregating it into clusters of events with similar information. This dissertation offers a complete data analysis of three OSINT feeds and the results that made us to design our solution, a detailed description of the architecture of our solution, its implementations and its validation, including the processing of events from other academic solutions.

Keywords: cibersecurity, open source intelligence (OSINT), indicators of compromise (IoCs), threat intelligence platforms (TIP), quality threat intelligence (TI), automated TI classification

Contents

List of Figures

List of Tables

List of Algorithms

Acronyms

AECCP	Automated Event Classification and Correlation Platform
API	Application Programming Interface
APT	Advanced Persistent Threats
AS	Autonomous System
CARO	Computer Antivirus Research Organization
CHIS	Covert Human Intelligence Sources
CIRCL	Computer Incident Response Center Luxembourg
CSIRT	Computer Security Incident Response Team
CSV	Comma-Separated Values
CTI	Cyber Threat Intelligence
ENISA	European Union Agency for Cybersecurity
ETIP	Enriching Threat Intelligence Platform
HTTP	Hypertext Transfer Protocol
HUMINT	Human Intelligence
ID	Identification/Identity/Identifier
IoC	Indicator of Compromise
IP	Internet Protocol
IT	Information Technology
MISP	Malware Information Sharing Platform
NATO	North Atlantic Treaty Organization
NCIRC TC	NATO Computer Incident Response Capability Technical Centre
NCSC-NL	National Cyber Security Centre Netherlands
OSINF	Open Source Information
OSINT	Open Source Intelligence
PURE	Platform for Quality Threat Intelligence
REST	Representational State Transfer
SANS	Sysadmin, Audit, Network, and Security
SIEM	Security Information and Event Management
SIGINT	Signals Intelligence
SOC	Security Operations Center
STIX	Structured Threat Information Expression
TECHINT	Technical Intelligence
TI	Threat Intelligence
TIP	Threat Intelligence Platform
TLP	Traffic Light Protocol
TOR	The Onion Router
TTP	Tactics, Techniques and Procedures
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
UT	Unified Taxonomy
VERIS	Vocabulary for Event Recording and Incident Sharing

Chapter 1

Introduction

In today's world, most of organizations are digital, operating with technologies and processes of the Internet era. The changes in IT infrastructure and usage models, including mobility, cloud computing, and virtualization have dissolved traditional enterprise security perimeters, creating a huge attack surface for hackers and other threat actors [1]. Managing the digital landscape in which an organization operates is a challenge that has never been more difficult, resulting in an organization vulnerable to many forms of attack.

Not only the digital landscape has evolved, but there has also been a significant evolution in cyber threat, as adversaries have advanced their knowledge. They have deployed increasingly sophisticated means of circumventing individual controls within users' local environments and probed further into their systems to execute well-planned and orchestrated attacks [2]. With the increase of the digital landscape and the threat landscape complexity, organizations are more likely to be targeted and suffer a severe cyber-attack, with high financial and reputational impact. The high likelihood and impact of cyber-attacks, in addition to the significant regulatory pressure to protect information, such as the European Union's General Data Protection Regulation, are encouraging organizations to look for new solutions to reduce their vulnerabilities [3].

One domain that has emerged during the past decade is cyber threat intelligence (CTI). This new domain combines key aspects from incident response and traditional intelligence and can be defined as "the process and product resulting from the interpretation of raw data into information that meets a requirement as it relates to the adversaries that have the intent, opportunity and capability to do harm" [4]. However, compared to other cyber domains, like incident response and security operations, CTI is still in the early adoption phase, limited by the lack of suitable technologies, known as threat intelligence platforms (TIPs) [1]. Despite organisations recognize the potentiality of CTI, the lack of tools that would help them manage the collected information and convert it to actions is preventing a mass adoption for this kind of solutions.

1.1 Motivation

With the emergence of new type of threat actors, like the advanced persistent threats (APTs), organizations cannot rely on a single solution to protect from this type of threats. The static approach of traditional security based on heuristic and signature does not match new threats that are known to be evasive, resilient and complex. These complex threats rely in well-advanced tactics to appear unknown to signature based tools and yet authentic enough to bypass spam filters [5]. To fight these threats, today's organizations must deploy a multi layered defence to improve their chances of detecting or disrupting an attack.

Cyber threat intelligence information, under a form of open source intelligence (OSINT), can provide knowledge to a vast selection of systems and processes that form this multi-layered defence, such as anti-virus and intrusion prevention systems and the processes that manage these solutions and review the events generated by them. This knowledge can be collected from many sources by using threat intelligence platforms (TIPs). However, TIPs receive thousands of security events, being hard to analyse them in order to extract relevant data about threats. According to recent surveys, the volume

and quality of data are the most common barriers to effective information exchange. Many interviewees report that, often, shared data is outdated and not specific enough to aid decision-making process, becoming unactionable [6]. The confidence level of information is another barrier since most sources do not provide this information, forcing analysts to put additional effort on evaluating and verifying the received data. Also, most organizations cannot make valuable use of their threat data because there is too much, approximately 250 to millions of indicators per day [6]. Considering the volume of shared threat information, most of the platforms end up being data warehouses rather than platforms where threat information can be analysed.

This dissertation proposes an approach to address some of the threat intelligence platforms limitations by generating highly information-rich objects under a common format and taxonomy defined by us and correlating and aggregating them into clusters of objects generating thus new threat intelligence with quality that share the same threat type and other information. Moreover, this study explores a solution to improve the response of threat analysts and all the systems used by organization against today's complex threats.

1.2 Objectives

This study aims at finding ways to benefit from OSINT to increase the detection capabilities of defence mechanisms, such as security information and event management systems (SIEMS) or intrusion detections systems (IDS), reducing the number of false positives and false negatives. In order to improve the collection of actionable cyber threat intelligence, we first need to understand the threat intelligence life cycle, the available information sources and current threat intelligence sharing platforms. This requires working on all levels of the intelligence gathering operation, using an automated system to receive data from multiple sources to improve the enrichment process and validate the information collected by cross referencing it and produce objects under a common format and taxonomy to store the obtained intelligence in such a way that it can be applied in the optimization of defence mechanisms. To achieve the overall goal of this project, two separate objectives must be completed.

The first objective that must be guaranteed is a clear understanding of the threat intelligence life cycle and of the current formats, taxonomies and platforms used in cyber threat intelligence sharing. This activity will allow us to determine which are the best available sources for information that can be converted into intelligence. It will also allow the selection and optimization of threat intelligence platforms to improve their efficiency. And finally, to determine if the existing formats and taxonomies for sharing threat intelligence are sufficient to store the intelligence produced or if they can be improved.

The second objective is the implementation of an infrastructure that will produce enriched intelligence objects, through a combination of sources, optimizing the configuration of the available platforms to extract the most from the available sources and developing a solution that can aggregate the processed information from different platforms into an enriched object.

1.3 Contributions

The main contributions of this dissertation are as follows:

- **Unified Taxonomy definition:** To reduce the overlapping of taxonomies with the same meaning, we propose a *single unified taxonomy*. This unified taxonomy is based on the eCSIRT.net incident taxonomy and CARO malware naming scheme and aims to simplify the event classification while maintaining its details.
- **Main attributes by threat:** To reduce the volume of shared information, we identified the most predominate attributes for each type of cyber threat based on analysis we preformed over CTI of different categories.

- **Overall design of our proposed solution:** We propose a solution that aims to improve threat intelligence quality produced by TIPs, by classifying and enriching it automatically. Our solution is composed of a set of smaller solutions (a `Classifier`, a `Trimmer`, an `Enricher` and a `Clusterer`), each one focused on one or more limitations verified in our data analysis.
- **Solution implementation:** We implemented our solution in the *Automated Event Classification and Correlation Platform*. Also, we describe the high-level implementation of our solution, following the architecture defined for each one of the modules of our platform (`Classifier`, `Trimmer`, `Enricher` and `Clusterer`). Moreover, we assessed our implementation with 3 OSINT feeds and events from other academic solutions.
- **Research statement:** The preliminary developed version of our solution gave rise to a research statement published at the Workshop on Data-Centric Dependability and Security (DCDS) 2019, entitled Generating Threat Intelligence by Classification and Association of Security Events [7].

1.4 Structure of the Document

This document if organized as follows:

- **Chapter 2:** Explains the context and related work of this dissertation, by introducing key terms and aspects for the project, which will allow to develop an understanding of the core elements of this research.
- **Chapter 3:** Presents all the data analysis performed in order to better understand the limitations of the TIPs and to project a solution to dead and minimize them.
- **Chapter 4:** Presents the overall design of our proposed solution, called Automated Event Classification and Correlation Platform (AECCP), which aims to improve threat intelligence quality produced by TIPs, by classifying and enriching it automatically.
- **Chapter 5:** Presents the high-level implementation of AECCP, following the architecture defined for each one of the modules of our platform (`Classifier`, `Trimmer`, `Enricher` and `Clusterer`).
- **Chapter 6:** Presents the evaluation of AECCP. This evaluation aims at validating AECCP ability to enrich, classify and correlate events, and evaluate each module it comprises.
- **Chapter 7:** Provides some remarks, presenting some limitations of our solution, some possible improvements and future work than can be done based on the results obtained.

Chapter 2

Context and Related Work

This chapter explains the context and related work of this dissertation, by introducing key terms and aspects for the project, which will allow to develop an understanding of the core elements of this research. The subject of advanced persistent threats will be briefly approached before diving into any other topic related to this research, as is the main challenge that pushes the development of today's new types of defence mechanisms, like threat intelligence platforms. To improve the understanding of threat intelligence, the concept of open source intelligence will first be introduced, followed by the definition of threat intelligence and its implementation in the context of cybersecurity with the use of indicators of compromise. The final element presented will be threat intelligence platforms to allow the understanding of how they work and to review currently available products.

2.1 Advanced Persistent Threats

Today's generation threats are multi-vectored, i.e., most attacks use multiple means of propagation, such as social engineering, email, and application vulnerabilities, and often multistage, meaning that most attacks operate in different phases, such as single device compromise, network lateral movement and data exfiltration [6]. These complex threats rely on social engineering techniques, the latest zero-day vulnerabilities, and well-advanced tactics for appearing unknown to signature-based tools and yet authentic enough to bypass spam filters. Traditional security defences were developed to inspect each attack vector as a separate path and each stage of an attack as an independent event, failing in identifying and analysing an attack as an orchestrated series of cyber incidents [5].

The advanced persistent threats (APT), being one of today's generation threats that had a major impact in the rise of cybercrime, branched from young hackers in the "black hat" community, whose objective was mayhem and reputation, to organized crime groups provided by states and private entities [1]. Ping Chen et al. proposed four characteristics to define advanced persistent threats and separate them from other criminal enterprises online, being them: specific targets and clear objectives, highly organized and well-resourced attackers, long-term campaigns with repeated attempts, and stealthy and evasive techniques [8].

- **Specific targets and clear objectives:** Targets are typically governments or organizations with significant intellectual property value. While traditional attacks propagate as broadly as possible to improve the chances of success, an advanced persistent threat attack only focuses on its pre-defined targets. As for the attack objectives, advanced persistent threats typically look for digital assets that bring competitive advantage or strategic benefits, such as intellectual property and trade secrets, while traditional threats mostly search for information that facilitates financial gain, like credit card data.
- **Highly organized and well-resourced attackers:** The actors behind advanced persistent threats are typically a group of skilled hackers, working in a coordinated way. They may work in a government cyber unit or be hired as cyber mercenaries by governments and private organizations. They are well-resourced from both financial and technical perspectives. This provides them with the ability to work for a long period, and have access to zero-day vulnerabilities and attack tools.

- **Long-term campaigns with repeated attempts:** An advanced persistent threat attack is typically a long-term campaign, which can stay undetected in the target's network for several months or years. Advanced persistent threat actors persistently attack their targets and they repeatedly adapt their efforts to complete the job. Traditional attackers often target a wide range of victims and move right on to something less secure if they cannot penetrate the initial target.
- **Stealthy and evasive techniques:** Advanced persistent threats attacks are stealthy, concealing themselves within enterprise network traffic, and interacting just enough to achieve the defined objectives. For example, APT actors may use encryption to obfuscate network traffic. This is different from traditional attacks, where the attackers typically employ tactics that alert the defenders.

2.1.1 Cyber Kill Chain

APTs can be understood from the defensive perspective of a "kill chain". Cyber kill chain is a model that defines a sequence of stages required for an attacker to successfully infiltrate a network and exfiltrate data from it. This model provides a framework to breakdown a complex attack into minor stages, enabling analysts to tackle smaller problems at the same time and helping the defenders to implement separate controls for each one of the phases. Cyber kill chain is mainly composed of seven stages, being them: reconnaissance, weaponize, delivery, exploitation, installation, command and control, and act on objective [9]. Figure 2.1 illustrates the sequence of these stages.

- **Stage 1 – Reconnaissance:** Information gathering (identification, selection and profiling) about a potential target. The information gathered from reconnaissance is used in later stages of cyber kill chain to design and deliver the payload. Reconnaissance is further divided into 2 types: passive reconnaissance – gathering the information about target without letting him know about it; and, active reconnaissance – deeper profiling of target which might trigger alerts.
- **Stage 2 – Weaponize:** Backdoor designing and a penetration plan, utilizing the information gathered from reconnaissance. Technically, the backdoor binds software exploits/vulnerabilities with a remote access tool, creating a silent backdoor capable of evading user attention and security mechanisms.
- **Stage 3 – Delivery:** Backdoor delivering, once again utilizing the information gathered from reconnaissance. Most deliveries require some kind of user interaction like downloading and executing malicious files or visiting malicious web pages on Internet. For delivering the weapon multiple delivery methods are used to increase the likelihood of delivery.
- **Stage 4 – Exploitation:** After delivering the cyber weapon, the next step is triggering the exploit. The objective of an exploit is to silently install the payload. To trigger the exploit there are certain conditions that need to be matched, such as the operating system and software versions, and the ability to avoid anti-virus or other security mechanism detection. For installing the payload multiple exploit are used to increase the likelihood of exploitation.
- **Stage 5 – Installation:** Malware nowadays are multi staged and heavily rely on advanced techniques to deliver the malware modules in a sophisticated manner. Before executing the core code, malware try to disable host-based security controls to continue undetected. Additionally, some malware instead of unpacking a large embedded copy of the core malware agent, they connect to a remote file repository to download the core components.
- **Stage 6 – Command and Control:** Command and Control (C&C) systems are used to give remote instructions to compromised machines. C&C systems can be centralized, peer-to-peer

decentralized or rely on a social network. Today's malware use techniques to hide communication patterns with its C&C. Anonymous communication techniques involve creating a channel resistant to traffic analysis, such as hiding data inside of media, using TOR network, using encrypted channels, etc.

- **Stage 7 – Act on objective:** After getting the communication setup with target system, the attacker executes the remote instructions based on its objective. This is an elaborate active attack process that takes months.

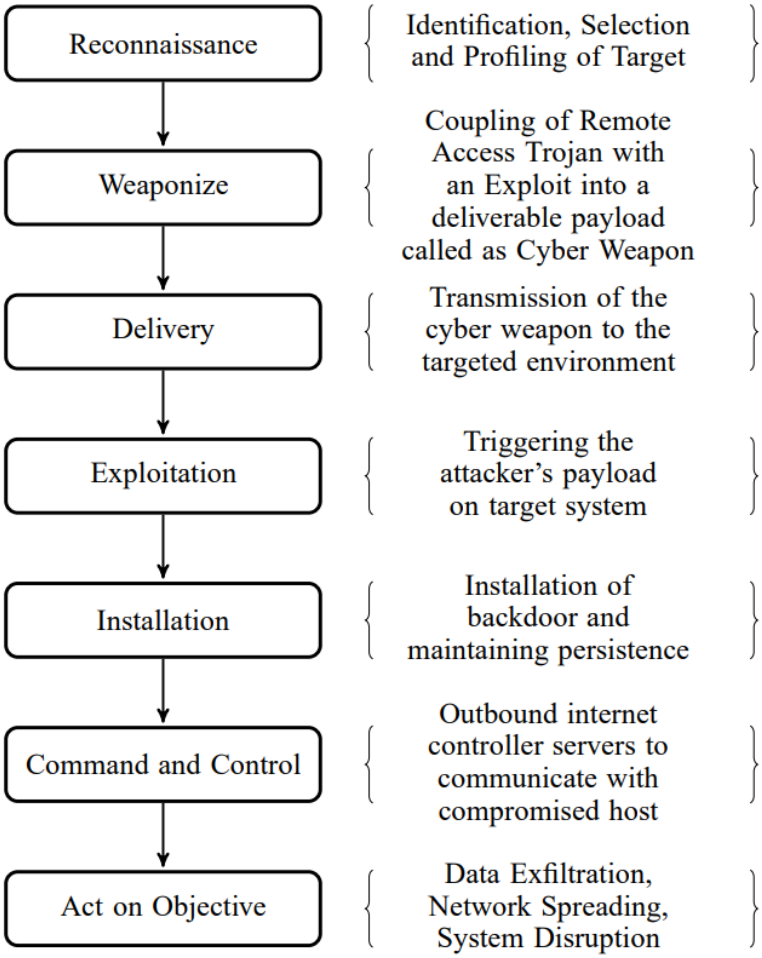

Figure 2.1: Phases of Cyber Kill Chain, adapted from [9]

2.2 Open Source Intelligence (OSINT)

The earliest forms of open source intelligence (OSINT) dates back to the Second World War, marked by the ability to find relevant information and combining it in a way that treats information as a resource rather than a commodity [10].

OSINT can be defined as intelligence produced from open source information (OSINF), that is, information that is publicly available. In other words, OSINF is information that is not confidential and is available in the public domain. It is the information that anyone can obtain by request, purchase, or observation. Examples of OSINF include the media (e.g., radio, television, newspapers, websites, blogs), official governmental reports, academic sources (e.g., papers, conferences, seminars), commercial data and so called 'gray literature' such as working papers, unofficial government documents and surveys. Nowadays, due to the development of the Internet, this type of information has

become significantly easier and cheaper to gather than the traditional public information acquired by clandestine services. In comparison to other sources of information, like human intelligence, OSINF can sometimes provide extra information and be a more reliable and safe way of acquiring intelligence [11].

To produce OSINT, OSINF is analysed, edited, filtered and validated. Moreover, the information gathered is linked with other sources, in order to verify, complement and contextualize the collected information. The more public available sources, the better intelligence will be produced. Figure 2.2 shows the transformation of data into information, via structure and context, then into intelligence, via analysis, as it flows through the intelligence cycle phases.

OSINT is one of the most common form of intelligence and considered a goldmine for the organizations [12]. One of the biggest advantages of using OSINT is the cost, as it is much less expensive compared to traditional information gathering tools. In addition to the cost advantage, OSINT has many advantages when it comes to sharing and accessing information, as information can be legally and easily shared with anyone, and open sources are always available and up to date [13].

However, OSINT has some constrains, such as the high quantity of available information that needs to be processed to create valid intelligence, therefore demanding an elevated quantity of work to extract useful information from the noise. This requires a large amount of analytical work from specialists in order to distinguish valid, verified information from false, misleading or inaccurate information. A final constrain of OSINT is that its production may not always provide the needed answer since it only uses the information that is available [13].

2.3 Threat Intelligence

Threat intelligence (TI) can be defined as “evidence-based knowledge, including context, mechanisms, indicators, implications and actionable advice, about an existing or emerging advice, about an existing or emerging menace or hazard to assets that can be used to inform decisions regarding the subject’s response to that menace or hazard” [14].

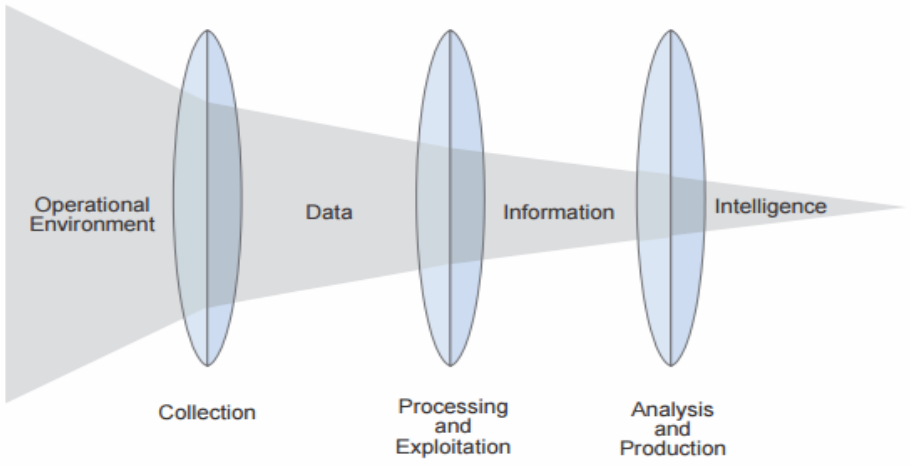

Figure 2.2: From data to intelligence [39]

In its simplest form, TI is the process of understanding the threats towards an organization based on available information. However, there must also be an understanding of how the information relates to the organization. Therefore, the information must be combined with contextual information to determine relevant threats to the organization. Furthermore, TI is useful to an organization only if it is actionable. If a team cannot determine how to best respond, combat or mitigate a threat to the organization, then the information provides little to no value [15]. Detecting incidents sooner, and potentially even preventing them, is the overall goal of TI. Organizations often see TI as a way to reinforce the environment and prepare for both known and unknown threats.

TI is growing in popularity and use amongst organizations of all sizes as organizations realize that attacks are increasing in size and complexity. According to 2020 SANS Cyber Threat Intelligence Survey, 85.5% of respondents have at least one person responsible to consume or produce TI in their organization and 7.1% of respondents plan to have one in the near future. This trend followed the evolution of targeted attacks and APTs as they require a different level of response that is more specific to the organization [16]. Many organizations are convinced that TI is one of the more valuable tools to help them better understand their attackers.

2.3.1 Threat Intelligence Cycle

The Intelligence Cycle is a five phase, continuous process to extract relevant intelligence in a timely manner to reduce risk and uncertainty. The five phases are: planning and direction; collection; processing and exploitation; analysis and production; dissemination and integration [17].

- **Planning and Direction:** Intelligence requirements and needs are identified based on the objectives for which the intelligence will be used. The level at which the intelligence will be required is one of the key elements that should be defined in this phase. There are three levels: strategic, operational and tactical [18].
  - *Strategic*: Information that allows to advise about risks and to improve decision making regarding cyber security investment.
  - *Operational*: Information focused on the motivation, intents and capabilities of the adversaries which may allow to predict their behaviour and next actions.
  - *Tactical*: Technical information that can be directly applied in the defence against attackers, such as IP addresses that can be used to define firewall rules to block the attacker's attempts

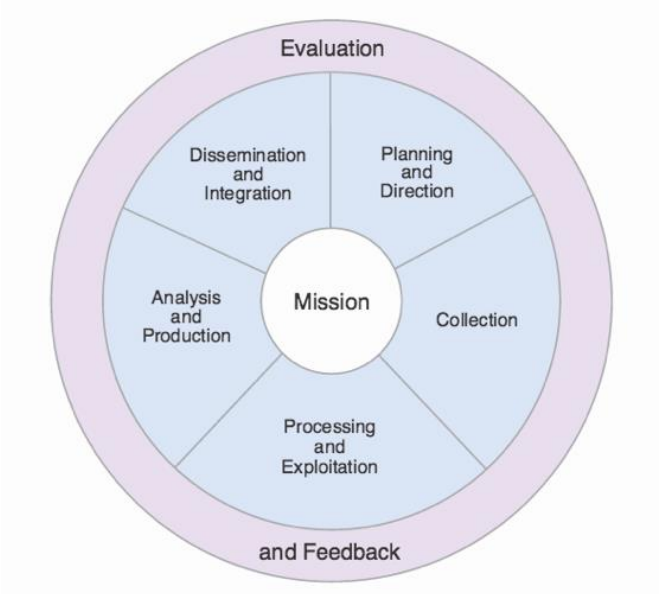

Figure 2.3: Intelligence cycle [39]

- **Collection:** All the activities related to the acquisition of the data necessary to satisfy the requirements are defined. Relevant data can be obtained from multiple sources and in multiple formats. Regarding the formats of intelligence, the most common come from human sources (HUMINT), sources that are exploited by concealed or covert means (CHIS), from publicly available sources (OSINT), from signal interceptions (SIGINT) and from technical sources like logs and malware analysis (TECHINT) [19].

- *Human Intelligence (HUMINT)*: intelligence derived overtly or covertly from human sources based on a relationship between an intelligence agent and the agent's handler;
- *Covert Human Intelligence Sources (CHIS)*: intelligence obtained by a person who establishes a relationship with another person for the covert purpose of using it to obtain or provide access to any information. It also includes intelligence derived from sources on the Deep Web that cannot be classified as OSINT since it is not public;
- *Open Source Intelligence (OSINT)*: intelligence derived overtly from publicly available sources;
- *Signals Intelligence (SIGINT)*: intelligence derived overtly or covertly from the interception of signals;
- *Technical Intelligence (TECHINT)*: although this is a variation of SIGINT it should not be confused with intelligence obtained 'by technical means' in that it does not involve any form of covert activity. One example of TECHINT is the logs generated routinely by hardware devices or software applications.

Regarding the sources, they can be grouped into two high-level categories, being them internal and external [15].

- *Internal TI:* Data points and information that are garnered from within the organization itself. The daily issues that can seem random and unconnected, can be organized into meaningful content by turning unrelated or simple events into intelligence. By logging details of the incidents, such as attack paths, vulnerabilities, malware and other network indicators, an organization can start to recognize similarities between incidents. Oftentimes, gathering internal information is much easier than organizing and interpreting it, due to the amount of data that are sent to a central aggregation point, such as a SIEM system.
- *External TI:* Intelligence that an organization acquires from outside itself. External TI can be further broken into multiple subgroups, namely data feeds, industry-specific groups, relationships with government and law enforcement, and crowdsourced platforms.

- **Processing and Exploitation:** The information collected in the previous phase is converted into a format that can be readily used. This implies parsing the collected data to identify the valuable parts, correlate the data obtained from different sources or moments in time, filter the noise, deduplicate and aggregate to reduce the quantity of data.
- **Analysis and Production:** The different pieces of information are transformed into a product that answers the requirements defined at the beginning of the cycle. This format can vary from a rule to be deployed in a firewall or intrusion prevention system, to an indicator of compromise (IoC).
- **Dissemination and Integration:** The intelligence that has been produced is delivered to and used by the target consumers, which can be internal or external to the organization. Traditionally, the distribution of intelligence was made through the traditional communication channels, like phone calls or emails. More recently, with the trending and evolution of TI, the distribution is made through specialized websites, automated distribution feeds and specialized platforms, known as threat intelligence platforms.

2.4 Standards and tools for exchange and processing TI

As previously stated, the objective of creating threat intelligence is the creation and delivery of a product that can be acted upon. While threat intelligence professionals find value in sharing threat information through informal and traditional communication channels, the results are inconsistent and unscalable.

To provide an adequate answer to today's complex threats, better frameworks were needed for communicating threat intelligence. Such frameworks should include: standardised reporting terminology and processes; benefit in information sharing for cyber security purposes; the ability for users to create trusted communities; and, a technical infrastructure to share and analyze threat intelligence at machine speed. In absence of an industry-standard framework, current sharing mechanisms include: private or restricted face-to-face meetings and phone calls; emails, forums and message boards; web portals with wiki-type capabilities; web portals acting as document management systems; web portals (some with APIs) allowing downloads of structured data; and, web portals offering social networking facilities with secure access and sharing controls [20].

2.4.1 Standard data formats

A lot of effort has already been put in structuring information for sharing purposes. According to a recent study, the most common standard to codify IoCs is STIX. However, its use is not widespread and poorly implemented [21].

Structured Threat Information Expression (STIX) is a language and serialization format used to exchange cyber threat intelligence (CTI). STIX enables organizations to share CTI with one another in a consistent and machine-readable manner, allowing security communities to better understand what computer-based attacks they are likely to see and to better prepare for and respond to those attacks faster and more effectively. STIX is designed to improve many different capabilities, such as collaborative threat analysis, automated threat exchange, automated detection and response. STIX provides an architecture based on 12 domain objects, that each represents a unique concept from CTI, that can be connected via relationships or sightings [22]. Figure 2.4 presents a schematic of the STIX architecture, followed by a brief description of each domain.

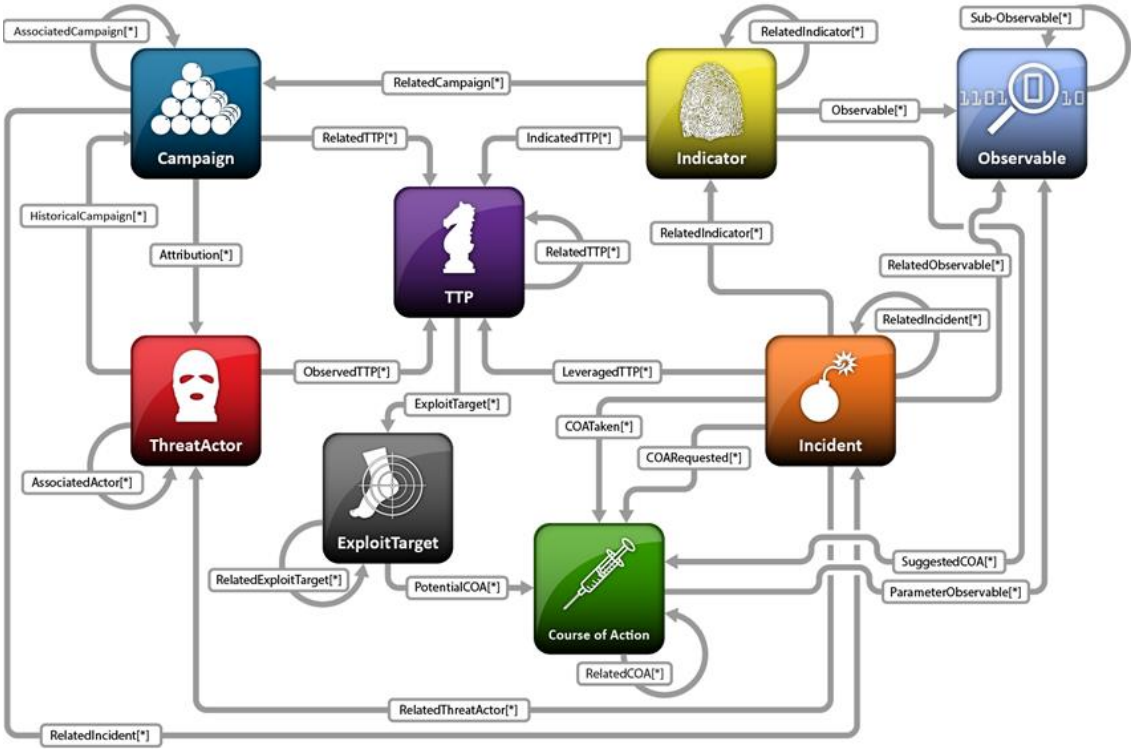

Figure 2.4: STIX architecture [22]

- **Attack Pattern:** A type of Tactics, Techniques, and Procedures (TTP) that describes ways threat actors attempt to compromise targets.

- **Campaign:** A grouping of adversarial behaviours that describes a set of malicious activities or attacks that occur over a period of time against a specific set of targets.
- **Course of Action:** An action taken to either prevent an attack or respond to an attack.
- **Identity:** Individuals, organizations, or groups, as well as classes of individuals, organizations, or groups.
- **Indicator**: Contains a pattern that can be used to detect suspicious or malicious cyber activity.
- **Intrusion Set**: A grouped set of adversarial behaviours and resources with common properties believed to be orchestrated by a single threat actor.
- **Malware**: A type of TTP, also known as malicious code and malicious software, used to compromise the confidentiality, integrity, or availability of a victim's data or system.
- **Observed Data:** Conveys information observed on a system or network (e.g., an IP address).
- **Report:** Collections of threat intelligence focused on one or more topics, such as a description of a threat actor, malware, or attack technique, including contextual details.
- **Threat Actor:** Individuals, groups, or organizations believed to be operating with malicious intent.
- **Tool:** Legitimate software that can be used by threat actors to perform attacks.
- **Vulnerability:** A mistake in software that can be directly used by a hacker to gain access to a system or network.

2.4.2 Threat Intelligence Sharing Platforms

In 2013, the concept of threat intelligence sharing platforms (in short threat intelligence platforms or TIPs) was introduced with the purpose of filling the industry-standard gap in threat intelligence sharing. TIPs usually vary in objective (some are used to operational information while others may be focused in long-term risk analysis), in scope of their action (from accepting only processed inputs to possessing natural language processing capacities) and in their capacities (current platforms range from data acquisition and storage to advanced analytics using machine learning). Despite their differences, the functionalities of the threat intelligence platforms follow the steps of the intelligence cycle. Most offer the following functionalities: collection and normalisation of machine readable feeds from multiple sources; correlation, pivoting and enrichment of data in order to add context; categorisation into indicators of compromise, threat actor type, geography, etc; integration of derived information into downstream security prevention and detection tools; co-ordination of the workflow of multiple users during incident response; and, sharing derived intelligence with other organisations at machine speed [23].

Based on the information obtained from the architecture and functionalities of diverse threat intelligence platforms, a generic TIP architecture was extrapolated and represented in Figure 2.5.

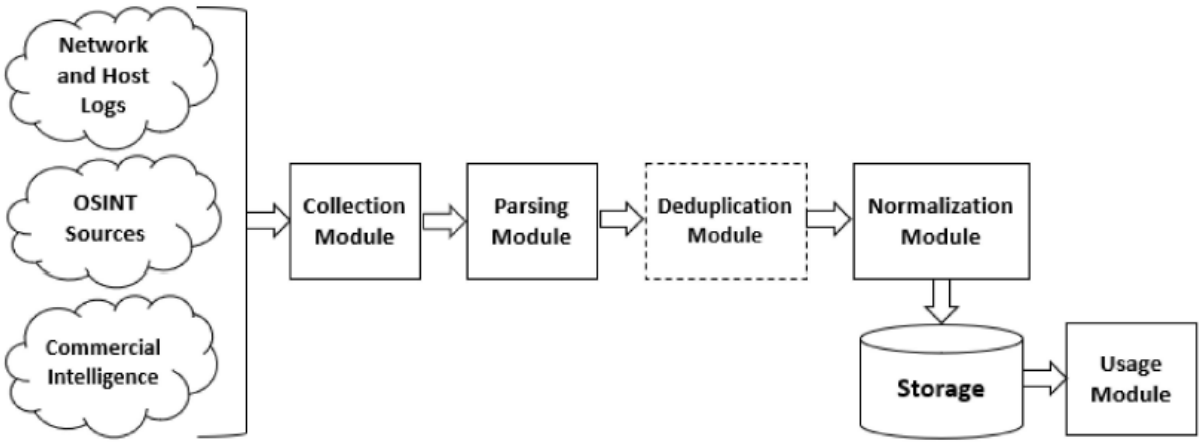

Figure 2.5: Generic TIP architecture [23]

A brief description of each module is presented below:

- **Collection Module:** The entry point of information into the platform from one or more sources of different types. The collection can be passive, configured to receive external feeds or with a functionality to allow the input of information by human users; or active, using a crawler to track content from specific locations, such as Twitter, blogs or forums.
- **Parsing Module:** Processes the collected sources to find keywords or specific text formats, such as IP addresses and hashes.
- **Deduplication Module:** Eliminates duplicates, therefore reducing the size of the processed information.
- **Normalization module:** Transforms the information into a standard format, to facilitate its processing and analysis.
- **Usage module:** Allows the consultation of the collected intelligence and its sharing with other participants. It may also allow for the conversion of the information into different standards, as well post-processing by its users.

2.4.3 Current limitations

TIPs have multiple advantages that enable organisations to easily bootstrap the core processes of collecting, normalising, enriching, correlating, analysing, disseminating and sharing of threat related information. However, current solutions have some limitations that prevents theirs mass adoption. Below are presented the limitations related to the current state and usage of TIPs [24].

- **Shared threat information is too voluminous:** One of the problems is the overload of threat information shared via open source, commercial sources and communities. Combining shared threat information from different sources makes the relevant intelligence hard to find and makes it difficult to generate value out of it.
- **Limited technology enablement in threat triage:** There is limited technology enablement to facilitate the relevancy determination process. Currently, this process is done manually, in a complex way and dependent on the analyst.
- **Focus on tactical indicators of compromise:** Tactical indicators of compromise are mostly shared lacking comprehensive threat information. During information sharing, standardized formats are underused or even not used, noting that most information is exchanged in unstructured files.

- **Focus on data collection:** Considering the volume of shared threat information and the limited analysis capabilities provided by TIPs, most of the platforms end up being data warehouses rather than platforms where threat information can be shared and analysed.
- **Trust related issues:** Most TIPs have limitations in the way that organisations interact and contribute to specific communities. Most platforms do not allow organisations to share only specific types of threat data with specific communities.
- **Data Quality:** Currently, the confidence level of information is not provided by most of the feed, forcing analysts to put additional effort on evaluating and verifying the received data.
- **Limited analysis capabilities:** Most TIPs have limited capabilities related to browsing, attribute-based filtering, advanced searched information, pivoting, exploration and visualisation. Moreover, few platforms provide integration with third party tools that could help addressing these limitations.
- **Diverse data formats:** While there are community efforts to provide connectors between different standards and formats, converting information without losing any elements or context from the source format is a challenge. Most TIPs tend to stay with one format, limiting the flexibility of the TIP users.
- **Limited advanced analytics capabilities and tasks automation:** Most TIPs have limited capabilities related to aggregation, composition, generalization as well as the capability to de-duplicate, automatically tag and classify data.
- **Shared intelligence without expiration date:** Currently, the time-to-live information is not provided by most of the feeds and TIPs have limited capabilities in handling this type of metadata information.
- **Diverse APIs and requirements for integration:** TIPs integrate with a (more or less) standard set of services and tools while requests for additional integrations are prioritized by the owners.
- **Limited workflow enablement:** Currently, TIPs provide limited workflow capabilities that would make the process of threat management more efficient, such as the capability of stakeholders to send requests for information.
- **Threat knowledge management limitations:** No common vocabulary is used for describing threat actors, tactics, techniques, procedures and tools.

2.5 MISP

Malware Information Sharing Platform (MISP) is a free and open source TIP initially created by the NATO Computer Incident Response Capability Technical Centre (NCIRC TC) as an implementation of the Smart Defence concept and, currently, owned by the Computer Incident Response Centre Luxembourg (CIRCL).

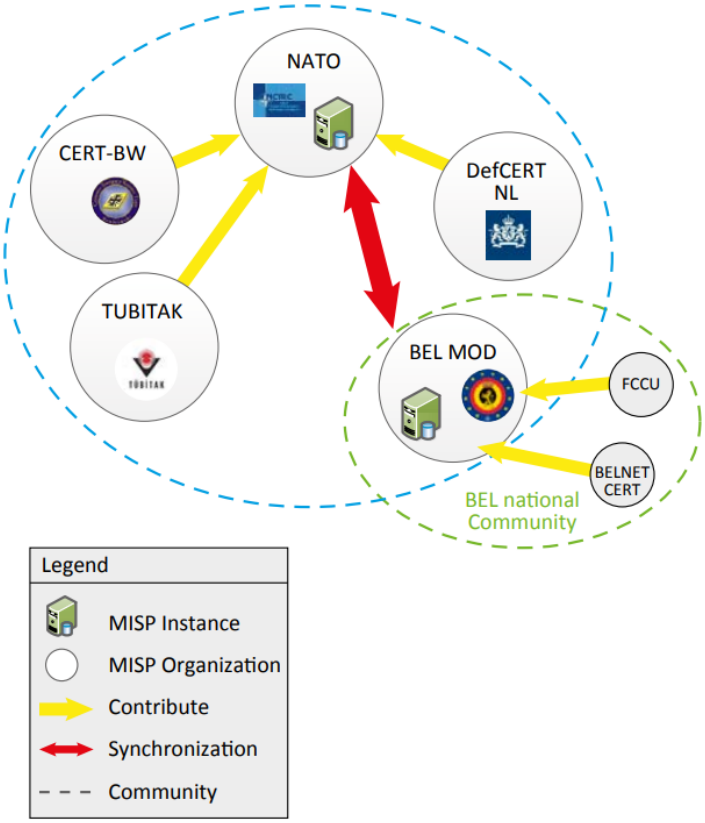

Figure 2.6: MISP communities [26]

One of the key concepts of MISP is the sharing of intelligence among members of the same community. MISP relies on the voluntary action of its community to share information and indicators, by leaving the decision of the to the sharer. Moreover, the sharer can select various sharing levels, ranging from sharing only within the organization to sharing within the whole MISP communities [25, 26, 27]. Figure 2.6 presents an example of MISP communities.

Regarding the capabilities, MISP already circumvents some of the limitations previously presented. Currently, MISP has not only, but mainly, the following capabilities: automatic correlation between indicators; sharing functionality with different models and levels of distribution; automatic exchange and synchronization of data among different MISP instances; advanced filtering capabilities; a graphical web interface to navigate seamlessly between indicators and their correlations; export of data in the most popular formats, namely STIX, OpenIOC, CSV and MISP standardized format; import of data in the most popular formats, as well free text to ease the integration of unstructured reports into MISP; proposal system to update indicators; flexible API to integrate MISP with other solutions; and, false-negative sighting and expiration sighting support [25].

2.5.1 Data model

As previously mentioned, MISP has its own format to exchange CTI. MISP standardized format allows users to decide the level of granularity of information to share, providing as much information as possible, or only the minimum of information for an event. MISP format has a flat model to ease the work of parsing and to avoid ambiguity, unlike STIX where observables are very often flattened and neglected by the parser which introduces rejected observables to be included [25]. Figure 2.7 presents a high-level representation of an MISP entry.

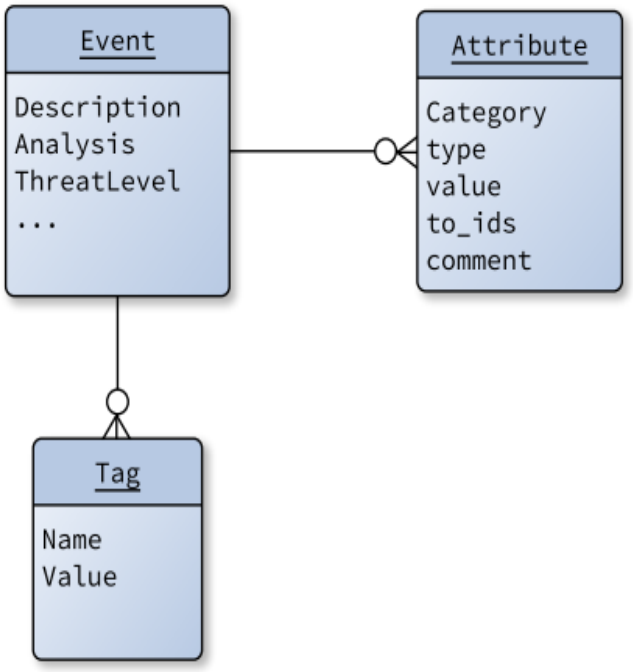

Figure 2.7: Simplified event representation in MISP [25]

A new entry in MISP is called an event object and can be defined as a set of characteristics and all kinds of descriptions of an IoC. These characteristics and relevant information are called attributes. Some examples for attribute types are: hash, filename, hostname and ip-address. An attribute can even be a complex object that contains multiple attributes. An example for a complex attribute is an anti-virus signature, which can contain the name of the anti-virus, the name of the signature and the detection date [25]. Furthermore, each attribute can be correlated with other simple or complex attributes. Figure 2.8 presents an example of an event of MISP with its attributes and connections.

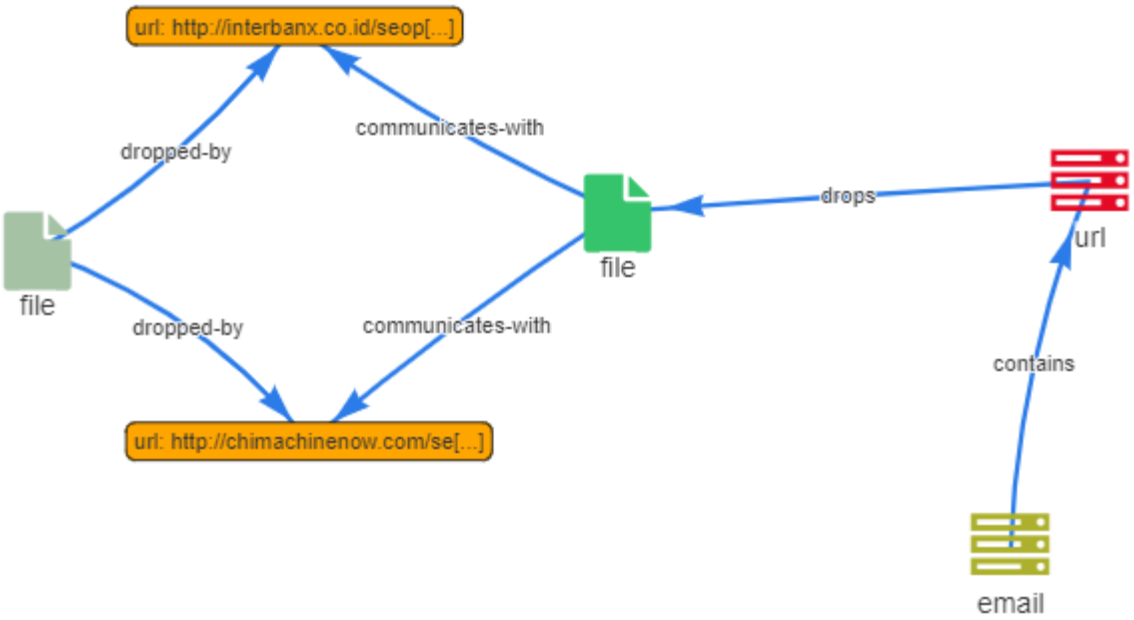

Figure 2.8: MISP event graph

2.5.2 Taxonomies

The classification of data is often bound to internal, community or national classification schemes. One common problem is the mapping of events into categories. This is a complex task since the number of categories is not always known in advance. Since a centralized pre-defined set of definitions that satisfies all the potential users is a hard challenge, MISP uses a distributed approach based on machine tags. However, the freedom of defining tags quickly lead to a situation where there were multiple tags with the same meaning making filtering complicated. To overcome this problem, a new concept of tagging was introduced, the taxonomies. A taxonomy is based on a triple tag structure with a namespace, a predicate and a value, for example, *enisa:nefarious-activity-abuse="ransomware"*. This flexible concept allows to classify and tag events following an organization own classification schemes or existing taxonomies used by other organisations. A clear advantage of this concept is the still human readable format of the machine tags [25].

In its default configuration, MISP includes a set of public incident classification schemes [28]. Here is the description of some of the most used schemes that will be referenced in the next chapters:

- **eCSIRT.net taxonomy** [29]**.** This taxonomy was developed many years ago, but the main categories are still current and can easily be used. On the other hand, the subcategories can lead to problems with how to classify an incident. Despite its defects, many European Computer Security Incident Response Teams (CSIRTs) use it, which give teams the opportunity to team up with others. Table 2.1 shows the main categories of the eCSIRT.net taxonomy in the MISP tag structure:

Table 2.1: eCSIRT.net taxonomy main categories

Tag
ecsirt:abusive-content
ecsirt:malicious-code
ecsirt:information-gathering
ecsirt:intrusion-attempts
ecsirt:intrusions
ecsirt:availability
ecsirt:information-content-security
ecsirt:fraud
ecsirt:vulnerable
ecsirt:other
ecsirt:test

- **CIRCL.LU taxonomy** [30]**.** MISP owner and main contributor uses its own taxonomy for classifying incidents. With some similarities with eCSIRT.net taxonomy, CIRCL.LU only has one level of classification. Table 2.2 shows the CIRCL.LU taxonomy in the MISP tag structure:

Table 2.2: CIRCL.LU taxonomy

Tag
circl:incident-classification="spam"
circl:incident-classification="system-compromise"
circl:incident-classification="scan"
circl:incident-classification="denial-of-service"
circl:incident-classification="copyright-issue"
circl:incident-classification="phishing"
circl:incident-classification="malware"
circl:incident-classification="XSS"
circl:incident-classification="vulnerability"
circl:incident-classification="fastflux"
circl:incident-classification="sql-injection"
circl:incident-classification="information-leak"
circl:incident-classification="scam"
circl:incident-classification="cryptojacking"
circl:incident-classification="locker"
circl:incident-classification="screenlocker"
circl:incident-classification="wiper"
circl:incident-classification="sextortion"

- **Microsoft implementation of CARO Naming Scheme** [31]**.** Microsoft designates malware and unwanted software according to the Computer Antivirus Research Organization (CARO) malware naming scheme. This scheme was created by a committee at CARO and was the first attempt to make malware naming consistent. Table 2.3 shows the Microsoft implementation of CARO Naming Scheme in the MISP tag structure:

Table 2.3: Microsoft implementation of CARO Naming Scheme

Tag
ms-caro-malware:malware-type="Adware"
ms-caro-malware:malware-type="Backdoor"
ms-caro-malware:malware-type="Behavior"
ms-caro-malware:malware-type="BroswerModifier"
ms-caro-malware:malware-type="Constructor"
ms-caro-malware:malware-type="DDoS"
ms-caro-malware:malware-type="Dialer"
ms-caro-malware:malware-type="DoS"
ms-caro-malware:malware-type="Exploit"
ms-caro-malware:malware-type="HackTool"
ms-caro-malware:malware-type="Joke"
ms-caro-malware:malware-type="Misleading"
ms-caro-malware:malware-type="MonitoringTool"
ms-caro-malware:malware-type="Program"
ms-caro-malware:malware-type="PUA"
ms-caro-malware:malware-type="PWS"
ms-caro-malware:malware-type="Ransom"
ms-caro-malware:malware-type="RemoteAccess"
ms-caro-malware:malware-type="Rogue"
ms-caro-malware:malware-type="SettingsModifier"
ms-caro-malware:malware-type="SoftwareBundler"
ms-caro-malware:malware-type="Spammer"
ms-caro-malware:malware-type="Spoofer"
ms-caro-malware:malware-type="Spyware"
ms-caro-malware:malware-type="Tool"
ms-caro-malware:malware-type="Trojan"
ms-caro-malware:malware-type="TrojanClicker"
ms-caro-malware:malware-type="TrojanDownloader"
ms-caro-malware:malware-type="TrojanDropper"
ms-caro-malware:malware-type="TrojanNotifier"
ms-caro-malware:malware-type="TrojanProxy"
ms-caro-malware:malware-type="TrojanSpy"
ms-caro-malware:malware-type="VirTool"
ms-caro-malware:malware-type="Virus"
ms-caro-malware:malware-type="Worm"

2.6 Related Work

In this section we present some relevant work developed in the Threat Intelligence field.

2.6.1 PURE

Platform for qUality thReat intelligencE, PURE, presented in “PURE: Generating Quality Threat Intelligence by Clustering and Correlating OSINT” is a platform that generates improved intelligence based on OSINT [32]. This improved intelligence translates into new enriched IoCs obtained by correlating and combining IoCs coming from different OSINT feeds that share information about the same threat. PURE uses a novel cluster method, the n-level correlation, for clustering correlated IoCs. This method allows the creation of clusters that can be summarized and converted into an enriched IoC, allowing the discovery of unidentified patterns and the detection of new complex attacks. PURE uses MISP to collect TI from OSINT feeds and other sources, such as TIPs. The feeds and the TIPs are channeled to receptors, which store IoCs as MISP events temporarily until they are processed. Pure can also use various TIPs (other than MISP) to take advantage of different capacities they have, such as the enrichment of OSINT by resorting to external information that does not come with it. The platform

comprises the normalization of the different IoC formats in a single one and compares the IoCs received with the IoCs stored in the database, using a metric of similarity that infers the existence of duplicates. It also discards IoCs that provide no new information and performs a filtering step over the single IoCs to create a threat of intelligence of quality. The set of IoCs of interest resulting from the filter is then sent to a clustering module, which applies similarity and weighs metrics over the IoCs of interest to aggregate similar and related IoCs. The attributes of the clusters created from the aggregation of similar and related IoCs are then correlated to find the most relevant information that characterizes a threat. Finally, PURE converts the cluster into a single enriched IoC as a MISP event and stores it in MISP database from which it can later be recovered.

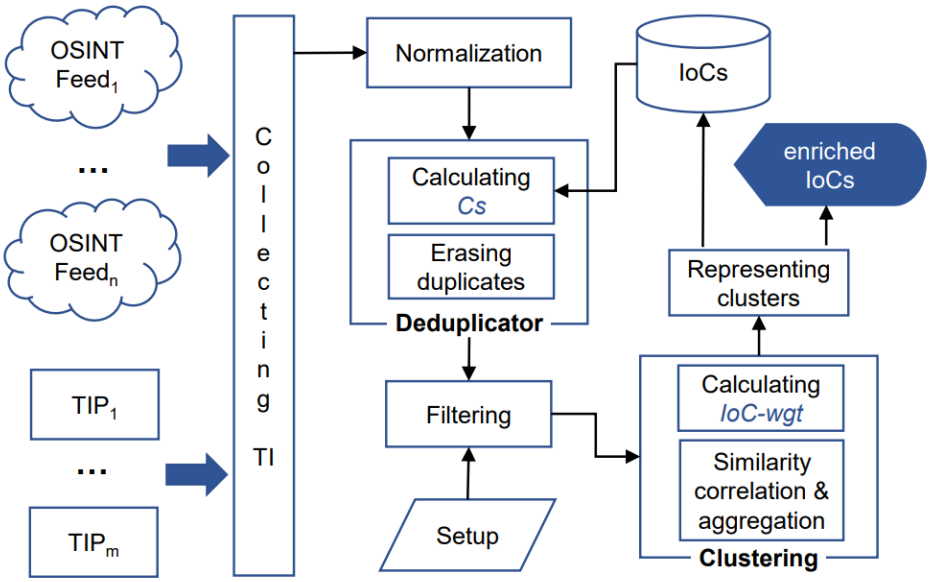

Figure 2.9 – PURE architecture [32]

2.6.2 ETIP

ETIP, an enriching threat intelligence platform presented in “Enriching Threat Intelligence Platforms Capabilities”, extends the importing capabilities, the quality assessment processes and the information sharing capabilities in current TIPs [33]. ETIP gathers and processes structured information from external sources, such as OSINT sources, and from a monitored infrastructure. ETIP is composed of two main modules: a composed IoC module, in charge of collecting, normalizing, processing and aggregating IoCs from OSINT feeds; and a context aware intelligence sharing module, able to correlate, assess and share static and real time information with data obtained from multiple OSINT sources. ETIP computes a threat score associated to each IoC before sharing it with other tools and trusted external parties. Enriched IoCs produced by ETIP contain a threat score that allows SOC analysts to prioritize the analysis of incidents. The threat score evaluates heuristics with two types of weights: individual weights assigned to every attribute based on their relevance, accuracy and variety, and; a global weight (i.e., completeness criterion) assigned to the heuristic. The higher the threat score value, the more reliable the IoC. [33]

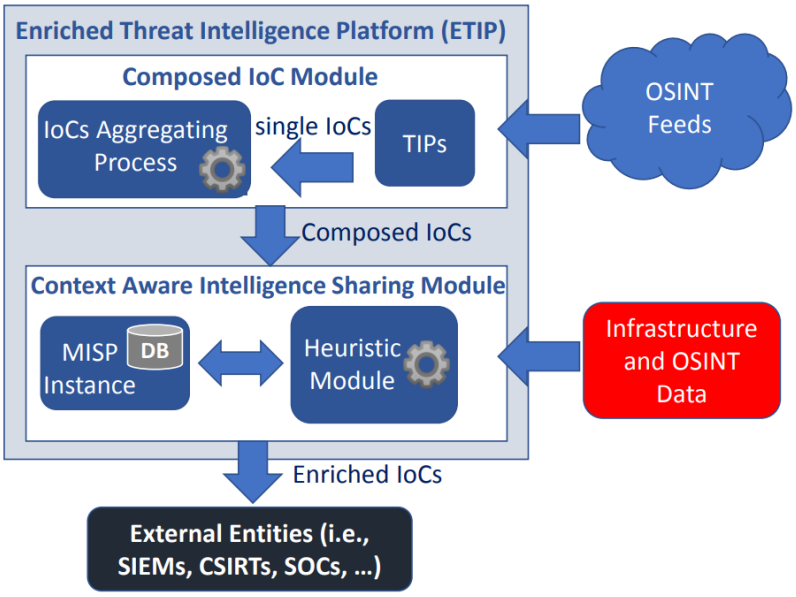

Figure 2.10 – ETIP architecture [33]

2.6.3 SYNAPSE

SYNAPSE, a Twitter-based streaming threat monitor for threat detection in security operation centres, implements a pipeline that gathers tweets from a set of accounts, filters them based on the monitored infrastructure, and classify the remaining tweets as either relevant or not. The pipeline is composed of a data collector, a filter, pre-processing and feature extraction module, a classifier, and a clustering module. The data collector requires a set of accounts, from which it will collect every posted tweet using Twitter's stream API. Despite the account-based collection approach, the collected data will include unrelated tweets which have to be dropped by a filter. The filtering approach assumes that a tweet referring a threat to a particular IT infrastructure asset must mention that asset. Only tweets that include at least one of the keywords will pass the filter. The pre-processing and feature extraction module is then used to normalise the tweet representation before proceeding to the Classifier. For the classification of tweets according to their security relevance, two classifiers were explored: Support Vector Machines and Multi-Layer Perceptron Neural Networks. Finally, SYNAPSE uses clustering to aggregate similar tweets in the news feed stream, using a Clustream algorithm adaptation to achieve the desired threat aggregation. Relevant tweets are grouped in dynamic clusters and presented as indicators of compromise that can be either manually inspected or fed to SIEMs and other threat intelligence tools. SYNAPSE tries to maximise relevant tweet information and minimise irrelevant tweet information before aggregating related tweets. [34]

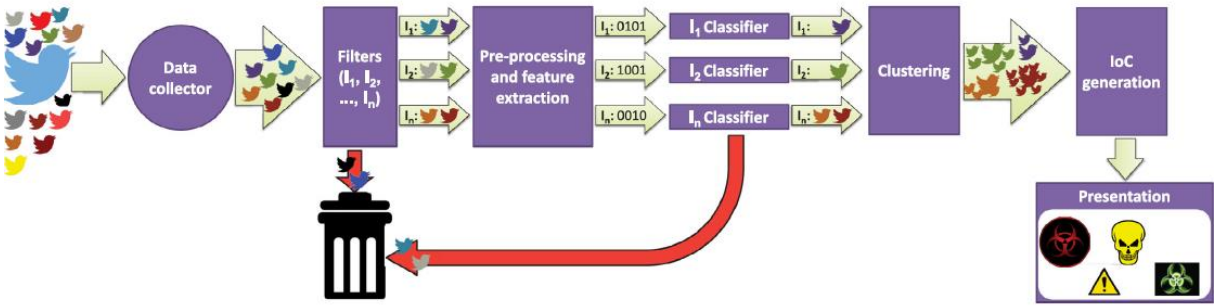

Figure 2.11 – SYNAPSE architecture [34]

Chapter 3

Data analysis for a Unified Taxonomy

This chapter presents all the data analysis performed in order to better understand the limitations of the TIPs and to project a solution to dead and minimize them. Each analysis was made having in mind the limitations of TIPs described in Chapter 2, Section 2.4.3, more specifically, limitations related to the processing of data in the platforms. Also, in this chapter, a description of the used data sources is provided, as well as the resulting dataset used in all the data analysis. Moreover, it presents an analysis over MISP taxonomies that shows how the vast set of public incident classification schemes included in MISP can increase unnecessary complexity and a single unified taxonomy proposed by us which can help to decrease it. In addition, an analysis over MISP attributes is provided, showing that too many attributes in a single event can increase unnecessary complexity, specifically if they do not add useful information, and a solution to face this problem is proposed. Finally, a brief explanation is given on how we can take advantage of references to external platforms.

3.1 Data source

As explained in Chapter 2, Section 4.2, every TIP needs to collect information in an active or passive way, however, to get the objectives of our work we did not need an active data collector. Thus, we opted to use external feeds as our source of information. However, we still had to choose which feeds we wanted, and we opted to use, as a starting point, the set of public OSINT feeds that MISP includes in its default configuration. In total, we had 50 feeds with different formats, namely MISP standardized format, CSV and free text feeds. CSV and free text feeds are only parsed as MISP Attributes and do not take advantage of all the MISP functionalities, in contrast to MISP formatted feeds that can be parsed from simple MISP Attributes to the more complex MISP Objects and benefit from all the MISP functionalities. Therefore, we left aside CSV and free text feeds and worked only with MISP formatted feeds, resulting in the following three feeds:

- **CIRCL OSINT Feed,** located at https://www.circl.lu/doc/misp/feed-osint/;
- **The Botvrij.eu Data,** located at http://www.botvrij.eu/data/feed-osint/;
- **inThreat OSINT Feed,** located at https://feeds.inthreat.com/osint/misp/.

From these three feeds, we were able to collect 1,366 events published by 14 different organisations. Figure 3.1 shows the distribution of the events according to their providers. Providers with less than or equal to 5 events were aggregated into “Other”, including, but not exclusively, VK-Intel, ESET and NCSC-NL.

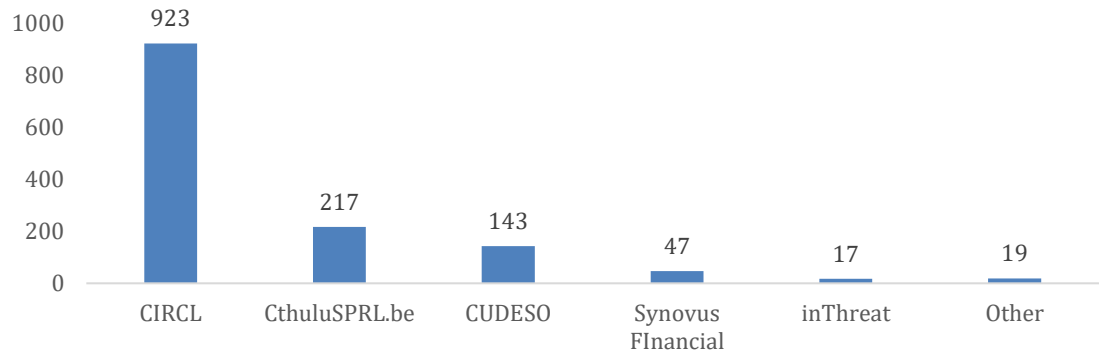

Figure 3.1: Distribution of events by provider

However, some of these events are dated to 2014, near the embryonic phase of the platform, meaning poorer events with minimal information and more events containing collections of IoCs from multiple attacks (e.g., blacklists). In contrast, recent events were richer in information and there were many more events corresponding to one single attack. Consequently, we shortened our dataset to only contain events from January 1st, 2016 until February 28th, 2019. In total, the data subset contained 1,168 events. Figure 3.2 shows the distribution of the events of the data subset according to their providers.

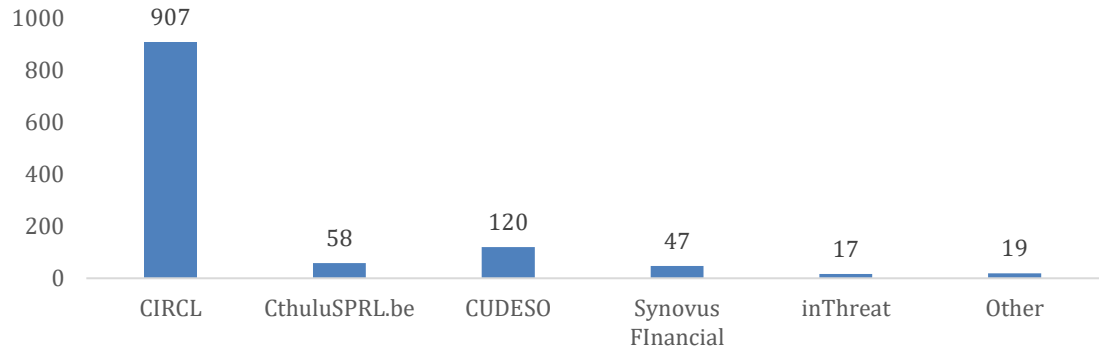

Figure 3.2: Distribution of events by provider started from January 2016 until February 2019

3.2 Unified Taxonomy Definition

Over the past decades, multiple cyber threat classification systems have been proposed, some of them focus on the classification of actors and methods [35], while others focus on specific techniques [36] or specific targets [37]. This complex array of taxonomies, with more than 100 classification systems, adds confusion when a threat is manually analysed by a threat analyst. In this section we present a simple solution to reduce this complexity by proposing a single unified taxonomy.

After the initial sizing of the dataset, a more detailed analysis was made in order to gather information about the number of classified events, more specifically events classified in accordance with a known incident classification taxonomy. As previously explained, MISP classifies events with tags, meaning that a classified event requires having at least one tag. Based on this principle, Figure 3.3 was created from analysis over tagged and untagged events.

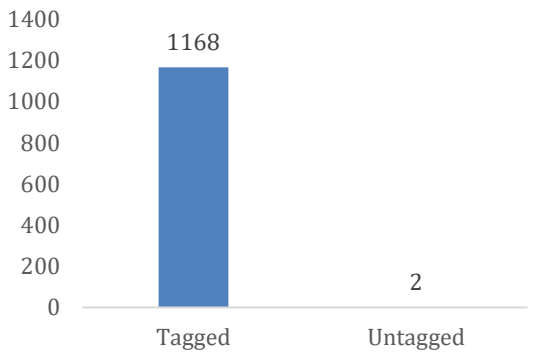

Figure 3.3: Tagged vs untagged events

Based on the previous analysis, we can conclude that almost every event is tagged. However, a more detailed analysis showed that many of the tagged events did not have a tag that allowed to classify them correctly. From the 1166 tagged events, 493 different tags were extracted. Table 3.1 shows the 10 most used tags in our dataset. A more extensive table can be found in Appendix A.

Table 3.1: 10 most used tags in events

Tag	Hits
tlp:white	1,133
osint:source-type="blog-post"	275
Type:OSINT	273
circl:incident-classification="malware"	218
malware_classification:malware-category="Ransomware"	113
ecsirt:malicious-code="ransomware"	98
misp-galaxy:ransomware="Locky"	70
inthreat:event-src="feed-osint"	32
osint:source-type="block-or-filter-list"	32
circl:topic="finance"	31

From the extracted tags, only 13% of them (62) corresponded to a known incident classification taxonomy, meaning that most tags did not add information about the type of the threat, but added information about its source and its sharing, such as the Traffic Light Protocol (TLP). Additionally, 61% of the tags corresponded to MISP Galaxies. MISP Galaxies are highly customizable and can correspond, not only to known attacks, but also to attack patterns, threat actors and tools. Therefore, we opted to not consider MISP Galaxy tags as classification tags. Due to the high heterogeneity and low information about the type of the threat, MISP Galaxy tags and “Other” tags were discarded from further analysis. Figure 3.4 shows the number of unique tags per their type.

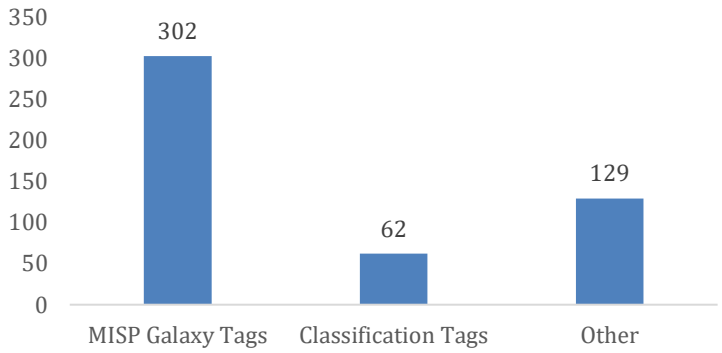

Figure 3.4: Types of extracted tags

From the 62 tags identified as classification tags, 10 different incident classification taxonomies were used, namely:

- CIRCL.LU taxonomy;
- eCSIRT.net incident taxonomy;
- ENISA threat taxonomy;
- ENISA threat taxonomy in the scope of securing smart airports;
- Europol common taxonomy for law enforcement and csirts;
- SANS malware classification based on “Malware 101 – Viruses” whitepaper;
- Microsoft implementation of CARO Naming Scheme;
- Vocabulary for Event Recording and Incident Sharing (VERIS);
- U.S. Democratic National Committee taxonomy;
- RiskIQ taxonomy.

Furthermore, we found that several events had multiple overlapping classification tags from different taxonomies. As an example, Table 3.2 maps different taxonomies related to ransomware, and which can appear in a same event.

Table 3.2: Mapping Table – Ransomware

Tag
ecsirt:malicious-code="ransomware"
enisa:nefarious-activity-abuse="ransomware"
malware_classification:malware-category="Ransomware"
ms-caro-malware:malware-type="Ransom"
veris:action:malware:variety="Ransomware”

Each of these events had corresponding classification tags, meaning duplicated information about their type. This taxonomy overload adds confusion when manually analysed since, most of the time, it creates unnecessary complexity, making the analysis harder and forcing the analyst to spend more time on it. In order to reduce this complexity, we propose a *single unified taxonomy*. This unified taxonomy is based on the eCSIRT.net incident taxonomy and CARO malware naming scheme and aims to simplify the event classification while maintaining its details. Since most taxonomies have two-tiers of classification, such as the CSIRT.net incident taxonomy, we opted to follow this level of detail. Moreover, this allows us to choose the granularity level of the classification. Table 3.3 contains an excerpt of our unified taxonomy, showing the relationship map we created to all public taxonomies included in MISP. The complete definition of the taxonomy can be found in Appendix B.

Table 3.3: Unified taxonomy (excerpt of) – public taxonomy mapping

Unified taxonomy		**Public taxonomies**
Tier1	**Tier2**	
abusive-content	spam	cccs:email-type="spam" circl:incident-classification="spam" ecsirt:abusive-content="spam" …
malware	adware	cccs:malware-category="adware" malware_classification:malware-category="adware" ms-caro-malware:malware-type="adware" …
	backdoor	maec-malware-behavior:maec-malware-behavior="install-backdoor" ms-caro-malware:malware-type="backdoor" ms-caro-malware-full:malware-type="backdoor" …

	browser-modifier	cccs:malware-category="browser-hijacker" ms-caro-malware:malware-type="broswermodifier" ms-caro-malware-full:malware-type="broswermodifier"
	cryptominer	circl:incident-classification="cryptojacking" maec-malware-behavior:maec-malware-behavior="mine-for-cryptocurrency" veris:action:malware:variety="click fraud"
	dialer	cert-xlm:malicious-code="dialer" ecsirt:malicious-code="dialer" ms-caro-malware:malware-type="dialer" …
	dos	maec-malware-behavior:maec-malware-behavior="denial-of-service" maec-malware-behavior:maec-malware-behavior="destroy-hardware" maec-malware-capabilities:maec-malware-capability="availability-violation" …
	exploit	cccs:malware-category="exploit-kit" enisa:nefarious-activity-abuse="exploits-exploit-kits" ms-caro-malware:malware-type="exploit" …
	hack-tool	ms-caro-malware:malware-type="hacktool"
	misleading	circl:incident-classification="screenlocker" enisa:nefarious-activity-abuse="rogue-security-software-rogueware-scareware" …
	monitoring-tool	maec-malware-behavior:maec-malware-behavior="capture' maec-malware-capabilities:maec-malware-capability="discovery" maec-malware-capabilities:maec-malware-capability="network-environment-probing" …
	password-stealer	enisa:nefarious-activity-abuse="credentials-stealing-trojans" maec-malware-behavior:maec-malware-behavior="crack-passwords" maec-malware-behavior:maec-malware-behavior="steal-password-hashes" …
	ransomware	cert-xlm:malicious-code="ransomware" cccs:malware-category="ransomware" circl:incident-classification="locker" …
	remote-access-tool	cccs:malware-category="webshell" enisa:nefarious-activity-abuse="remote-access-tool" enisa:nefarious-activity-abuse="botnets-remote-activity" …
	settings-modifier	ecsirt:malicious-code="malware-configuration" ms-caro-malware:malware-type="settingsmodifier" ms-caro-malware-full:malware-type="settingsmodifier”
	spammer	maec-malware-capabilities:maec-malware-capability="email-spam" maec-malware-behavior:maec-malware-behavior="send-email-message" ms-caro-malware:malware-type="spammer" …
	spoofer	ms-caro-malware:malware-type="spoofer" ms-caro-malware-full:malware-type="spoofer"
	spyware	cccs:malware-category="keylogger" cert-xlm:malicious-code="spyware-rat" cccs:malware-category="spyware" …
	trojan	cert-xlm:malicious-code="trojan-malware" cccs:malware-category="trojan" ecsirt:malicious-code="trojan" ..
	virtool	cert-xlm:malicious-code="rootkit" cccs:malware-category="rootkit" ecsirt:malicious-code="rootkit" …
	virus	cert-xlm:malicious-code="virus" cccs:malware-category="virus" ecsirt:malicious-code="virus" ..

	wiper	circl:incident-classification="wiper" veris:action:malware:variety="destroy data" maec-malware-behavior:maec-malware-behavior="erase-data" …
	worm	cert-xlm:malicious-code="worm" cccs:malware-category="worm" ecsirt:malicious-code="worm" …
information-gathering	scanning	cert-xlm:information-gathering="scanner" circl:incident-classification="scan" ecsirt:information-gathering="scanner" …
	sniffing	cert-xlm:information-gathering="sniffing" ecsirt:information-gathering="sniffing" pentest:network="sniffing" …
	social-engineering	cert-xlm:information-gathering="social-engineering" ecsirt:information-gathering="social-engineering" enisa:nefarious-activity-abuse="social-engineering" …
intrusion-or-attempts	ids-alert	cert-xlm:intrusion-attempts="exploit-known-vuln" ecsirt:intrusion-attempts="ids-alert" europol-event:brute-force-attempt …
	brute-force	cert-xlm:intrusion-attempts="login-attempts" ecsirt:intrusion-attempts="brute-force" europol-event:brute-force-attempt …
	unknown-exploit	cccs:exploitation-technique="other" cert-xlm:intrusion-attempts="new-attack-signature" ecsirt:intrusion-attempts="exploit"
	account-compromise	cert-xlm:intrusion="privileged-account-compromise" cert-xlm:intrusion="unprivileged-account-compromise" ecsirt:intrusions="privileged-account-compromise" …
	system-or-application-compromise	cert-xlm:intrusion="application-compromise" cert-xlm:intrusion="domain-compromise" circl:incident-classification="sql-injection" …
	botnet-member	cert-xlm:intrusion="botnet-member" ecsirt:intrusions="bot"
availability	dos-or-ddos	cccs:event="dos" circl:incident-classification="denial-of-service" csirt_case_classification:incident-category="DOS" …
information-content-security	unauthorised-information-access	cert-xlm:information-content-security="unauthorised-information-access" common-taxonomy:information-security="unauthorised-access" ecsirt:information-content-security="unauthorised-information-access" …
	unauthorised-information-modification	cert-xlm:information-content-security="unauthorised-information-modification" common-taxonomy:information-security="unauthorised-modification-or-deletion" ecsirt:information-content-security="unauthorised-information-modification" …
fraud	masquerade	cert-xlm:fraud="masquerade" ecsirt:fraud="masquerade" enisa:nefarious-activity-abuse="identity-theft-identity-fraud-account” …
	phishing	cccs:email-type="phishing" cccs:event="phishing" circl:incident-classification="phishing" …
vulnerable	vulnerable-service	cccs:misusage-type="vulnerable-software" cert-xml:vulnerable="vulnerable-service" ecsirt:vulnerable="vulnerable-service" …

Additionally, a bag of words was defined for each category in the unified taxonomy to describe them and allowing further classification. Each bag was created based on words from the public taxonomies, and synonyms from these extracted words. These bags of words will not only support further analyses over events with public taxonomy tags, but most importantly be used to analyse events without public taxonomy tags, as for example those that were not classified yet. Table 3.4 presents the unified taxonomy from Table 3.3 mapped with bag of words, by category.

Table 3.4: Unified taxonomy – bag of words

Unified taxonomy		**Words**
Tier1	**Tier2**	
abusive-content	spam	'spam', 'junk email', 'junk mail', 'junk e-mail', 'unsolicited email', 'unsolicited mail', 'unsolicited e-mail', 'bulk email', 'bulk mail', 'bulk e-mail', 'unwanted email', 'unwanted mail', 'unwanted e-mail'
malware	adware	‘adware’
	backdoor	‘backdoor’
	browser-modifier	'browser hijacker', 'browser modifier'
	cryptominer	'cryptominer', 'cryptojacking', 'cryptomining', 'cryptojacker', 'miner', 'mining'
	dialer	‘dialer’
	dos	'dos', 'ddos', 'destruction', 'destroy', 'destroying'
	exploit	'exploit'
	hack-tool	'hacktool', 'hack tool'
	misleading	'joke', 'misleading', 'rogue', 'rogueware', 'scareware', 'screenlocker'
	monitoring-tool	'monitoring', 'monitor', 'scanning', 'scanner', 'sniffing', 'sniffer', 'probe', 'probing'
	password-stealer	'password stealer', 'credential stealer', 'password theft', 'credential theft', 'password stealing', 'credential stealing'
	ransomware	'ransom', 'ransomware'
	remote-access-tool	'remote access'
	settings-modifier	'settings modifier', 'setting modifier', 'configuration modifier', 'configurations modifier'
	spammer	'spammer', 'spam'
	spoofer	'spoofer', 'spoofing'
	spyware	'spyware', 'keylogger'
	trojan	'trojan', 'trojanclicker', 'trojandownloader', 'trojandropper', ‘clicker', 'downloader', 'dropper'
	virtool	'rootkit', 'rootkits', 'virtool'
	virus	'virus', 'viruses'
	wiper	'wiper', 'erasure', 'erase', 'wipe', 'wiping', 'erasing'
	worm	'worm', 'worms'
	scanning	'scanning', 'scan', 'scanner'

information-gathering	sniffing	'wiretapping', 'monitoring'
	social-engineering	'social', 'engineering', 'personnel behaviour', 'impersonation', 'impersonations', 'impersonating', 'trick', 'tricks', 'tricking', 'deception', 'deceptions', 'elicitation'
intrusion-or-attempts	ids-alert	'attempt to compromise', 'attempted compromise', 'attempt to exploit', 'attempted exploit', 'attempt exploitation'
	brute-force	'brute', 'login attempt', 'login attempts'
	unknown-exploit	'unknown exploit', 'new attack', 'new signature'
	account-compromise	'account compromise', 'credentials compromise', 'successful login', 'login with success', 'authenticated with success', 'successful authentication'
	system-or-application-compromise	'domain compromise', 'application compromise', 'system compromise', 'domain intrusion', 'application intrusion', 'system intrusion'
	botnet-member	'bot', 'botnet member'
availability	dos-or-ddos	'dos', 'ddos', 'denial of service', 'disruption', 'degradation', 'exhaustion'
information-content-security	unauthorised-information-access	'unauthorised access', 'unauthorised information access', 'unauthorised data access'
	unauthorised-information-modification	'unauthorised modification', 'unauthorised information modification', 'unauthorised data modification'
fraud	masquerade	'masquerade', 'forged identity'
	phishing	'phishing', 'pharming', 'spearphishing', 'whaling'
vulnerable	vulnerable-service	'vulnerable', 'vulnerability'

3.3 Threat main attributes

As previously stated, the volume of shared information is one of the TIPs' limitations. This limitation was observed during the analysis of our dataset in the following formats:

- **Events containing collections of IoCs from multiple attacks.** Most of these events contain IoCs with few or none correlations. For example, some of these events contain lists of malicious IPs with the main purpose to serve as an input for a detection or prevention component. Since these events contain long lists of attributes with few to none context between each other, we opted to discard them from further analyses, in order to not negatively impact our results. In total, 17 events were discarded from the 1168 events.
- **Events with too many attributes.** 20% of our dataset contained events with more than 100 attributes. From the point of view of a SOC analyst, the more attributes an event has, the more difficult it is to analyse

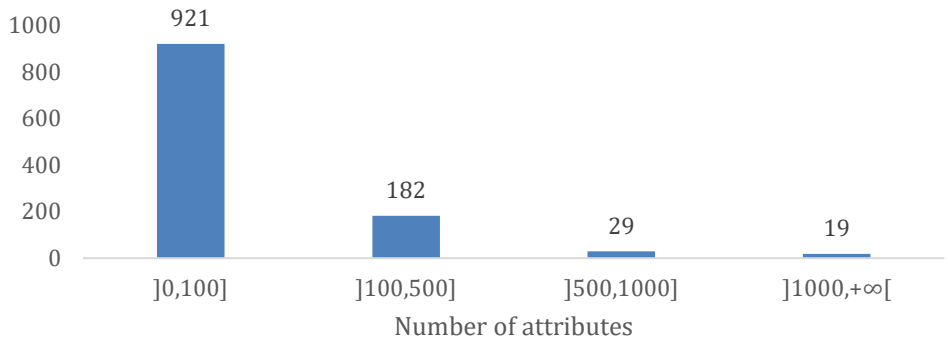

Figure 3.5: Events per number of attributes

In order to support the analysis focused on the events with too many attributes, preceding analyses were needed. These analyses combined the results by the number of attributes, in order to differentiate the results from smaller events and bigger events. For this purpose, 4 intervals were used:]0,100],]100,500],]500,1000] and]1000,+∞[.

The first preceding analysis was a more granular data analysis based on Figure 3.5 results. This analysis was supported by Appendix B public taxonomies' tags, in order to classify each event according to our unified taxonomy. More precisely, each tag from each event was compared with the public tags and, when matched, classified according to the corresponding `Tier1` category of our unified taxonomy. Figure 3.6 gives an overview of the number of events that we were able to classify. As we can observe, many events were not classified (460 out of 1151), because they did not have any classification tags, and so did not match with any taxonomy. It is important to note that some events were classified with more than one `Tier1` category, because they had more than one public tag, and they corresponded to different unified taxonomy categories. Figure 3.7 shows the 691 classified events for `Tier1` categories.

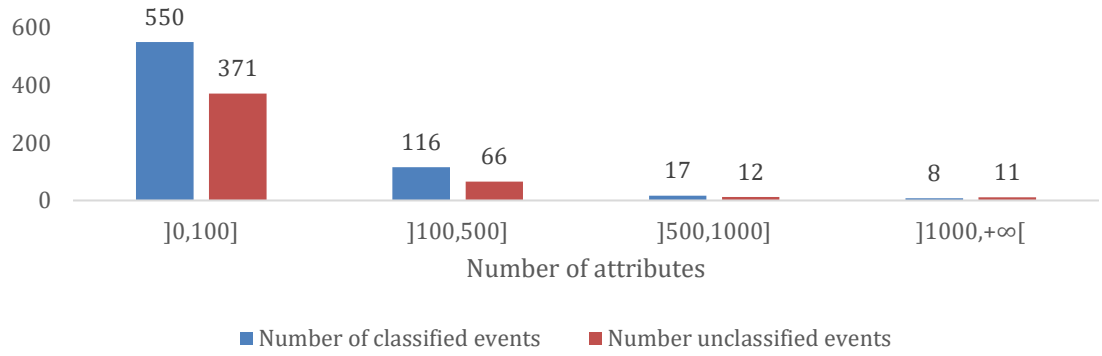

Figure 3.6: Classified events per number of attributes

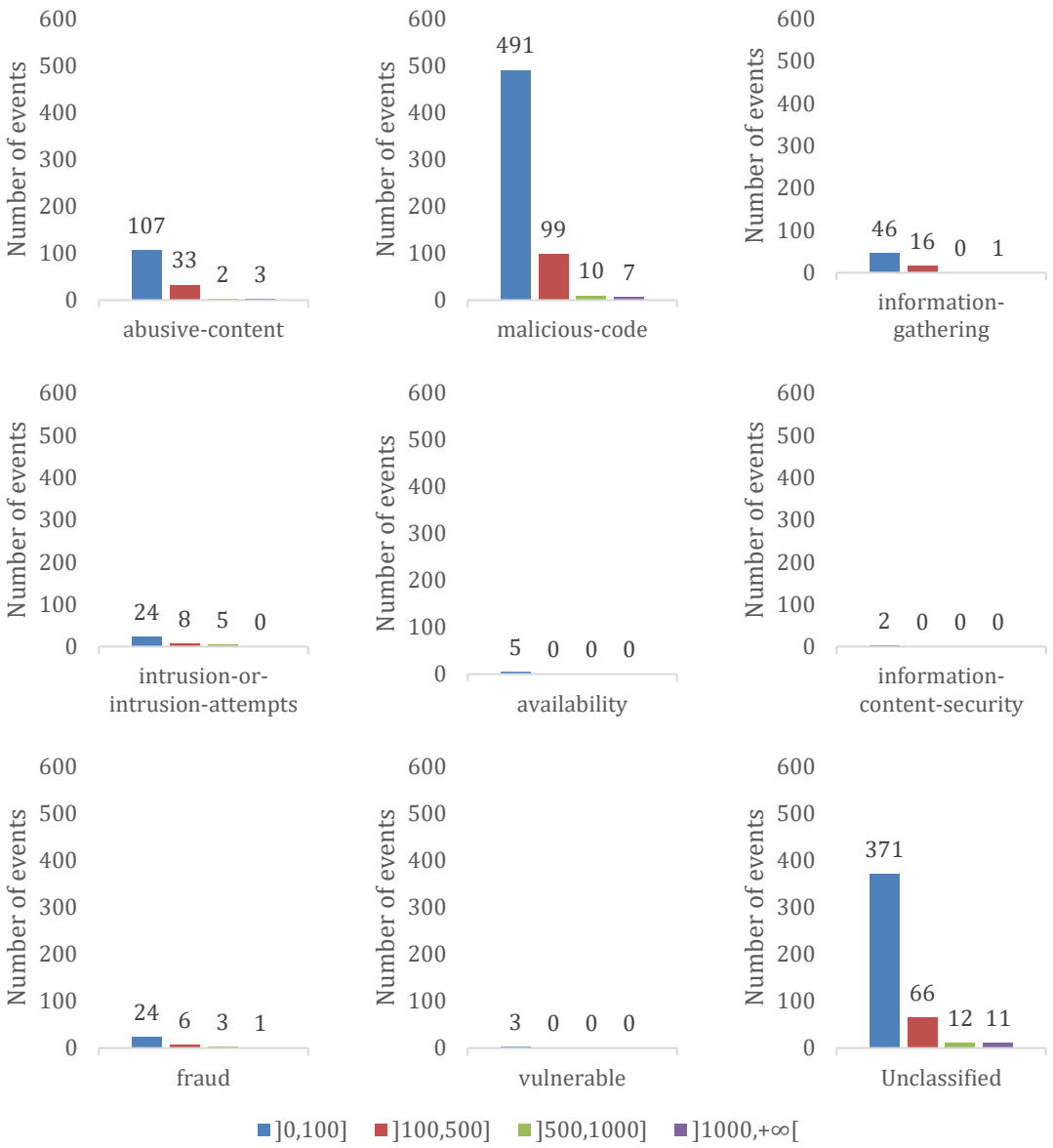

Figure 3.7: Classified events per number of attributes and tier1 category

The results from Figure 3.7 indicate that our dataset was low on events with tags related to the following `Tier1` categories: availability, information-content-security and vulnerable. Since we have few events from these categories, the results associated with them from subsequent analyses will not be considered in the development and evaluation of our work.

Due to the high amount of MISP supported attribute types, a second analysis was made in order to identify attributes with similar properties. For example, both MD5 and SHA1 are hash values that are used as a checksum to verify data integrity, so they will be aggregated into the same group named “file hash”. By aggregating attributes with similar properties, the results from the analyses will be focused on the characteristics of the attributes and not only on their type, meaning that, even if our dataset only have attributes with the type MD5, attributes with the type SHA1 will not be discarded from the results, since they belongs to the same group. Table 3.5 contains the attribute types supported by MISP and their corresponding group, defined by a manual analysis over the characteristics of every attribute type.

Table 3.5: Attribute groups

Group	Attribute type
Agent	email-x-mailer, user-agent
Bank account	btc, cc-number, iban, xmr, bank-account-nr,
Bank id	aba-rtn, bic, bin
Bank other info	payment-details
Date	datetime, whois-creation-date, date-of-birth, issue-date-of-the-visa,
Email address	dns-soa-email, email-dst, email-reply-to, email-src, target-email, whois-registrant-email
Email name	email-dst-display-name, email-src-display-name
Email other info	email-header, email-message-id, email-mime-boundary, email-thread-index
Email text	email-body, email-subject
File hash	authentihash, cdhash, filename|authentihash, filename|impfuzzy, filename|imphash, filename|md5, filename|pehash, filename|sha1, filename|sha224, filename|sha256, filename|sha384, filename|sha512, filename|sha512/224, filename|sha512/256, filename|ssdeep, filename|tlsh, impfuzzy, imphash, md5, pehash, sha1, sha224, sha256, sha384, sha512, sha512/224, sha512/256, ssdeep, tlsh
File name	email-attachment, filename
File other info	malware-type, mime-type, mobile-application-id, pdb
File sample	malware-sample
Location	country-of-residence, nationality, passport-country, place-of-birth, primary-residence
Mac address	mac-address, mac-eui-64
Network address	ip-dst, ip-dst|port, ip-src, ip-src|port, port, target-machine
Network hash	hassh-md5, hasshserver-md5, ja3-fingerprint-md5
Network id	AS
Network name	domain, domain|ip, hostname, hostname|port
Network request	http-method, cookie
Organization	whois-registrant-org, whois-registrar, target-external, target-org
Other Info	anonymised, Boolean, comment, counter, float, github-organisation, github-repository, github-username, hex, link, other, size-in-bytes, text, attachment
Pattern	pattern-in-traffic, pattern-in-file, stix2-pattern, pattern-in-memory
Personal id	frequent-flyer-number, identity-card-number, jabber-id, passenger-name-record-locator-number, passport-number, redress-number, target-user, twitter-id, visa-number
Personal location	target-location, travel-details
Personal name	first-name, last-name, middle-name, whois-registrant-name
Personal other info	gender, special-service-request
Phone number	phone-number, whois-registrant-phone, prtn
Process name	windows-scheduled-task, windows-service-displayname, windows-service-name
Process other info	named pipe, mutex
Regkey	Regkey, regkey|value
Rule	bro, sigma, snort, yara, zeek
Threat actor	campaign-id, campaign-name, threat-actor
Travel	place-port-of-clearance, place-port-of-onward-foreign-destination, place-port-of-original-embarkation
URI	uri
URL	url
Vulnerability	cpe, vulnerability
X509 fingerprint	x509-fingerprint-md5, x509-fingerprint-sha1, x509-fingerprint-sha256

Based on the preceding analyses, an analysis focused on the events with too many attributes was made. This analysis had the objective to identify the most predominant attribute groups for each `Tier1` category. Since the events with more attributes have a higher impact on the results, due to the weight of an event being directly proportional of the amount of the attributes in itself, the 4 intervals previously

used were considered. The following tables (Tables 3.6-3.13) show the most predominant attribute types for each `Tier1` category. The complete tables can be found in Appendix C.

Table 3.6: Most predominant attributes for abusive-content

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[
URL	30%	25%	22%	17%
Network address	28%	26%	29%	25%
Network name	27%	23%	20%	16%
File hash	8%	14%	15%	23%
Other Info	3%	6%	6%	8%
File sample	2%	6%	6%	11%
File name	2%	1%	1%	0%
Email text	1%	0%	0%	0%

Table 3.7: Most predominant attributes for malicious-code

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[
File hash	24%	29%	33%	32%
URL	17%	15%	13%	10%
Network address	17%	16%	15%	13%
Network name	16%	15%	13%	21%
Other Info	15%	16%	16%	15%
File name	3%	3%	3%	2%
Date	2%	2%	2%	2%
File sample	1%	2%	2%	4%
Email address	1%	1%	0%	0%
Bank account	1%	1%	0%	0%
Regkey	1%	0%	0%	0%
Rule	1%	0%	0%	0%

Table 3.8: Most predominant attributes for information-gathering

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[
Network address	35%	25%	25%	13%
File hash	22%	23%	23%	11%
Other Info	12%	10%	10%	5%
URL	12%	12%	12%	6%
Network name	12%	23%	23%	61%
File name	2%	3%	3%	2%
Vulnerability	1%	0%	0%	0%
Email text	1%	0%	0%	0%

Table 3.9: Most predominant attributes for intrusion-or-intrusion-attempts

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[
Other Info	31%	23%	10%	10%
File hash	30%	31%	13%	13%
Network name	22%	7%	6%	6%
Date	7%	7%	3%	3%
File name	4%	3%	1%	1%
Network address	3%	27%	54%	54%
URL	3%	2%	11%	11%
Email address	1%	0%	0%	0%

Table 3.10: Most predominant attributes for availability

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[
Network name	33%	33%	33%	33%
Network address	25%	25%	25%	25%
Other Info	23%	23%	23%	23%
File hash	14%	14%	14%	14%
Rule	2%	2%	2%	2%
Date	1%	1%	1%	1%
File name	1%	1%	1%	1%
URL	1%	1%	1%	1%

Table 3.11: Most predominant attributes for information-content-security

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[
Other Info	52%	52%	52%	52%
File name	29%	29%	29%	29%
File hash	11%	11%	11%	11%
Date	3%	3%	3%	3%
File sample	1%	1%	1%	1%
Network address	1%	1%	1%	1%
Regkey	1%	1%	1%	1%
URL	1%	1%	1%	1%

Table 3.12: Most predominant attributes for fraud

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[
Network name	50%	49%	58%	81%
File hash	14%	23%	13%	6%
URL	11%	4%	5%	2%
Other Info	11%	9%	11%	5%
Email address	5%	1%	3%	1%
Network address	4%	5%	3%	2%
Rule	2%	1%	0%	0%
File name	1%	3%	2%	1%
Vulnerability	1%	0%	0%	0%

Table 3.13: Most predominant attributes for vulnerable

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[
File hash	53%	53%	53%	53%
Other Info	18%	18%	18%	18%
File name	13%	13%	13%	13%
Network name	11%	11%	11%	11%
Rule	3%	3%	3%	3%
Network address	2%	2%	2%	2%
Process other info	1%	1%	1%	1%

As previously mentioned, the events with more attributes have a higher impact on the statistical analysis, since the weight of an event corresponds to the amount of its attributes. This can be confirmed from the results presented in the Tables 3.6 to 3.13. As a result, when the analysis was performed over all the classified events (]0,+∞[interval), some of the results had great discrepancy compared to the result from an analysis restricted to events with less than 100 attributes. For example, in Table 3.8 the attributes group “network name” equals 12% of all groups when the analysis is only made over events with less than 100 attributes, and the same attributes group equals 61% of all groups when including all the classified events in the analysis. Although, the results from Figure 3.7 show that almost 80% of our classified dataset is formed by events with less than 100 attributes, these events have less weight in comparison to the remaining 20% of our classified dataset. Moreover, even though we have much less

events in the]100, +∞[interval than in the]0, 100] interval, the]100, +∞[interval creates higher impact in the results than the]0, 100] interval. Since our dataset is composed mainly of events with less than 100 attributes, we have higher trust in the results gathered from those. Thus, we opted to use the result from the]0,100] interval. This information will be used to improve the global quality of the events by only using the most important attributes of each category.

3.4 OSINT references to external platforms

Another key finding from our dataset was the large amount of references to external platforms in the form of links. More than 90% of the links pointed to VirusTotal [38], an online service that analyse files and URLs enabling the detection of viruses, worms, trojans and other kinds of malicious content using antivirus engines and website scanners. Additionally, these platforms like VirusTotal tend to provide APIs allowing to access information without using the website interface. However, the amount of these references increases the time that an analyst requires to analyse the event since the analyst needs to jump between platforms to gather information and also process it manually. We consider this as a TIP's limitation (not pinpointed on Chapter 2, Section 4.3) which can easily be turned into a benefit and it is considered in our proposed solution (see Chapter 4).

Chapter 4

Automated Event Classification and Correlation Platform

This chapter presents the overall design of our proposed solution, called *Automated Event Classification and Correlation Platform* (AECCP), which aims to improve threat intelligence quality produced by TIPs, by classifying and enriching it automatically. In practice, our solution is composed of a set of smaller solutions, each one focused on one or more limitations verified in our data analysis detailed in Chapter 3 and some of those presented in Chapter 2. Regarding the limitation related to the volume of shared information, we propose an approach to reduce the number of attributes per event based on the most predominant attributes of its category. Moreover, regarding incident taxonomy management, we propose an approach to classify every event according to the unified taxonomy defined in Chapter 3. Since this solution will analyse and classify events in an automated way, it also increase technology enablement in threat triage. Furthermore, we propose a solution to enrich the data quality of an event based on OSINT from VirusTotal platform. Finally, in order to increase the advanced analytics capabilities of MISP, we propose a solution that creates new events as clusters of enriched events from the same category and with related attributes in common, after a correlation process that looks for relationships between attributes of different events.

Table 4.1 depicts the limitations that we addressed in AECCP as well as the proposed solution for each one and in which section they are presented. However, for a better understanding of the approaches, Section 4.1, presents the symbolic representation of an event that is used along the sections, and Section 4.2 gives an overview of the platform, showing the workflow and interactions between its components.

Table 4.1: Addressed limitations and correspondent proposed solutions

Limitation	Proposed approach	Section
Threat knowledge management limitations	Every event will be classified according to the unified taxonomy defined in Chapter 3, Section 3.2.	4.3
Limited technology enablement in threat triage	The classification of each event will be automated, based on its data (description of the attack, anti-virus reports, etc.)	
Shared threat information is too voluminous	Each event will have a simplified view only containing the most predominant attributes stated in Section 3.3, of Chapter 3.	4.4
Data Quality	Events containing links to VirusTotal will be enriched with information provided by the platform. Additionally, events containing hashes and URLs will also be enriched using the same method.	4.5
Limited advanced analytics capabilities and tasks automation	The classification of each event will be automated based on its data (description of the attack, anti-virus reports, etc.)	4.3
	When at least two events from the same category have an attribute in common, a cluster will be created in order to help an analyst identify related events.	4.6

4.1 Symbolic representation of an event

Along this chapter we will use a generic and simplified representation of events as shown in Figure 4.1, to facilitate and better understand the details of the approaches. This simplified representation contains the ID of the event, it's description, tags, attributes and the relations between those attributes within the event. The ID of the event is characterized as E_x being x a variable. The tags are characterized as T ranging from 1 to n. Tags from the unified taxonomy have a u attached (^{u}T). Moreover, an event can have no tags, meaning that the value of this field can be null. Furthermore, the attributes of an event are characterized as A also ranging from 1 to m. Attributes enriched with information (e.g., from VirusTotal) have an e attached (^{e}A). Additionally, the relations between attributes will be represented using a hyphen. For example, A_1 - A_2 represents a relation between A_1 and A_2 attributes. Finally, all the other data of an event with minor relevance for this work will be compact into the field "other data". In brief, the following legend will be used to represent an event in this chapter:

- E_x – Event X
- $E_{x'}$ – Modified Event X
- T – Tag
- ^{u}T – Unified Taxonomy Tag
- A – Attribute
- ^{e}A – Enriched Attribute

E_x	
Description	
Other data	
Tags	$T_1 \dots T_n \mid {}^{u}T_1 \dots {}^{u}T_j \mid$ NULL
Attributes	$A_1 \dots A_m \mid {}^{u}A_1 \dots {}^{u}A_n$
Relations	A_i - $A_j \dots A_x$ - $A_v \mid {}^{u}A_a$ - ${}^{u}A_b \dots {}^{u}A_y$ - ${}^{u}A_z \mid$ NULL

Figure 4.1: Generic and simplified representation of an event

4.2 AECCP Overview

AECCP is a platform that interacts with TIPs (e.g., MISP) in order to classify, enrich and correlate the events received by them. Moreover, all AECCP work is automated based on the results of the analyses made in Chapter 3.

As previously explained, our solution is composed of a set of smaller solutions. More specifically, AECCP is composed of 4 modules: a `Classifier`, a `Trimmer`, an `Enricher` and a `Clusterer`. The `Classifier`, detailed in Section 4.3, aims at classifying each event according to the unified taxonomy. The `Trimmer`, detailed in Section 4.4, aims at reducing the volume of the attributes in an event, based on the relevancy of those attributes. The `Enricher`, detailed in Section 4.5, aims at enriching the events with information from VirusTotal. At last, the `Clusterer`, detailed in Section 4.6, aims at creating clusters of events that share the same category and have at least an attribute in common.

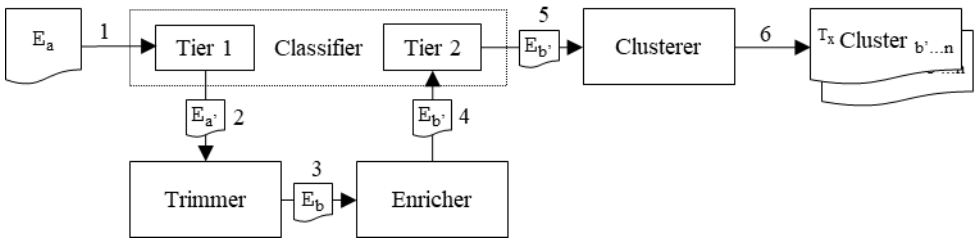

Figure 4.2: Representation of the interactions between modules

In order for the final solution to be most efficient, each module will interact in a specific pipeline. This pipeline assures that each event has the most information required for each module before being processed by it. Figure 4.2 represents the proposed interactions between modules, explained bellow according to the numbers in the figure:

1. E_a, a source event from MISP database, is served as input to the `Classifier` without any pre-processing from our proposed solution. To get the most accurate classification, E_a is firstly only classified according to the Tier 1 category of the unified taxonomy. The Tier 1 ^{u}T is then added in the source event tag list transforming it to $E_{a'}$. No new event is created in this step. However, if E_a could not be classified according to the Tier 1 ^{u}T due to lack of information, the event proceeds without a Tier 1 ^{u}T.
2. The `Trimmer` module iterates over the event attributes. In order to reduce the workload of the following modules, the `Trimmer` precedes the `Enricher`. The `Trimmer` receives $E_{a'}$ (the event transformed by the `Classifier`), as input and creates $E_{b'}$ a new event with the most relevant attributes and all ^{u}T from $E_{a'}$. Based on a threshold defined by the SOC analyst and the Tier 1 ^{u}T of E_b, the most predominant attributes of that categories are then copied from $E_{a'}$ to E_b if their relevancy percentage stay above the defined threshold. In case $E_{a'}$ has no Tier 1 ^{u}T, $E_{a'}$ is processed the same way as if $E_{a'}$ had all Tier 1 ^{u}T to not lose any predominant attributes.
3. E_b, the new event created by the `Trimmer`, is processed by the `Enricher`, witch as `Trimmer` acts over attributes. In this module, attributes in the event containing URLs or hashes are updated with information from the VirusTotal, transforming E_b to $E_{b'}$. Additionally, the `Enricher` adds an associated enriched attribute to $E_{b'}$ for each $E_{b'}$ attribute that was updated (enriched). This new attribute will support the output of antivirus engines, website scanners and analysis tool (that allowed the update).
4. $E_{b'}$, the event updated by the `Enricher` is reprocessed by the `Classifier`, this time according to the Tier 2 category of the unified taxonomy. Since the event was enriched with information not existent in the beginning of the processing, from the `Enricher`, the `Classifier` is able to classify the event more accurately. In this step the Tier 1 ^{u}T are updated with Tier 2 ^{u}T (e.g., $^{u}T_1$: $^{u}T_2$). Events that could not be classified according to Tier 1 category in step 1 are reprocessed and classified according Tier 1 and Tier 2 categories. If it still could not be classified, the event exits the pipeline and is not processed by the further modules.
5. $E_{b'}$, the event updated by the `Enricher` and the `Classifier`, is served as an input to the `Clusterer`. In this module, other events that share at least one Tier 2 ^{u}T with $E_{b'}$ and have at least one valuable attribute (attributes that provide context to a specific attack, i.e., hashes) in common with $E_{b'}$ are clustered in a new event $^{Tx}Cluster_{b'...n}$. Moreover, this module is

recursive, meaning that it tries to find other events related to every event added to the cluster. Additionally, multiple new events ($^{uTx}Cluster_{b',...n}$) can be created by the `Clusterer`, if $E_{b'}$ has more than one Tier 2 category tag.

4.3 Automated event classification

As explained in Chapter 3, Section 3.2, the high diversity of classification tags can be a disadvantage from the point of view of threat knowledge management. Additionally, due to this diversity, most events must be manually analysed to identify their categories. Since most threat triage and periodization processes rely on the category of the event, this manual process can create unwanted delay in the subsequent processes. In order to reduce both of these limitations, we propose a `Classifier`.

The `Classifier` automatically classifies events received by the platform according to the unified taxonomy, based on the tag, description and attribute information of the events. More specifically, it classifies events using two methods: *classification based on public taxonomies tags* and *classification based on keywords*.

Regarding the first method, classification based on public taxonomies tags, the `Classifier` takes advantage of the mapping information from Table 3.3 to update every public taxonomy tag to our unified taxonomy. In other words, each event served as an input to the `Classifier` will have its tags scanned and matched against the unified taxonomy mapping table. When matched, the corresponding unified taxonomy tag is added to the event tags list, if not already in the list. For example, if an event has the tags `cert-xlm:information-gathering="scanner"` and `circl:incident-classification="scan"`, the unified taxonomy category tag `unified:information-gathering=″scanning″` will be added to the event tag list once.

Regarding the second method, classification based on keywords, the `Classifier` uses the bag of words from Table 4.4 to identify keywords related to a unified taxonomy category based on the information contained in the description, attributes and custom tags (tags that do not belong to a public taxonomy) of the events. As we previously mentioned, some events hold important details in their descriptions that can help an analyst to identify the category of the incident. Moreover, it is also possible to gather important information from attributes and custom tags of an event to better classify it. In other words, each event served as an input to the `Classifier` will also have its custom tags, description and attributes scanned and matched against the bag of words defined in Table 4.4. When matched, the related unified taxonomy tag is added to the event tags list, if not already in the list. Opposed to the first method, this method can classify events that were not tagged yet (i.e., without classification tags). For example, if the word `phishing` is found in the description of an event with no tags, the event will be updated to contain the tag `unified:fraud=″phishing″` in its tag list.

Figure 4.3 shows the transformation of an event E_a processed by the `Classifier`. When E_a is processed using the first method, tags T_1 to T_n are scanned and matched against the unified taxonomy mapping table. When matched, the corresponding unified taxonomy tag $^{u}T_x$ is added to $E_{a'}$. However, if there are no tags in E_a tag list, no unified taxonomy tags will be added using this method. Regarding the second method, when E_a is processed, its description, as well its custom tags from T_1 to T_n and its attributes from A_1 to A_n are scanned to identify keywords that match the bag of words defined for each unified taxonomy category. When matched, the corresponding unified taxonomy tag $^{u}T_x$ is added to $E_{a'}$.

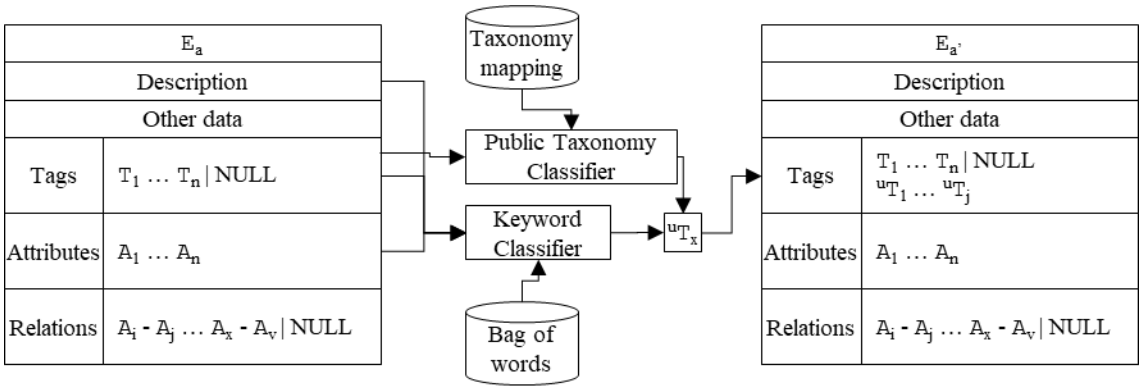

Figure 4.3: Representation of an event processed by the Classifier

As seen in Figure 4.3, each event is processed two times by the `Classifier` in AECCP, in step 1 and step 4, each time according a different unified taxonomy Tier.

In step 1, the `Classifier` classifies $E_{a'}$ according to Tier1. During this step, the `Classifier` uses the two methods described above (*classification based on public taxonomies tags* and *classification based on keywords*) based on information from E_a. In this step only Tier 1 ^uT (e.g., $^uT_1$) tags are added to $E_{a'}$.

Finally, in step 4, the `Classifier` updates the ^uT added in step 1 to $E_{a'}$, but now according to Tier2. As we will see in Sections 4.4 and 4.5, $E_{a'}$ suffers additional changes to enrich its information before being reprocessed by the `Classifier`, resulting in $E_{b'}$. During step 4, the `Classifier` uses the same two methods as in step 1 but with some minor changes. When using the *classification based on public taxonomies tags* the `Classifier` uses the tag list from $E_{a'}$, since $E_{b'}$ lost its public taxonomy tags in the `Trimmer`, as explained in Section 4.4. When using the *classification based on keywords* the `Classifier` uses $E_{b'}$, since this event was enriched in the `Enricher`, as explained in Section 4.5, therefore containing more information to be consumed by the `Classifier`. In step 4, the $^uT_1$ tags are updated with Tier 2 tags (e.g., $^uT_1$:$^uT_2$).

4.4 Event simplification

The amount of shared information derived from events with too many attributes was another limitation verified in Chapter 3, in Section 3.3. Both manual and automated analysis of events are impacted by unnecessary information. This type of information mainly acts as “good to know”, in opposite to “need to know”, creating noise and consequently adding complexity to the event. In order to minimize this limitation, we propose a `Trimmer`.

The `Trimmer` automatically trims the less relevant attributes from events based on their unified taxonomy Tier 1 category and according to the predominant attributes (i.e., “good to know” information) resulting from the analysis presented in Section 3.3 – Tables 3.6 to 3.13. Each event served as an input to the `Trimmer` will have its attributes scanned and “classified” according to the attribute groups defined in Table 3.5. Afterwards, based on a global relevancy threshold defined by the SOC analyst, for example 10%, for each attribute, if it belongs to a group with lower relevance than the relevancy threshold (based on the analysis performed on Chapter 3, Section 3), the attribute is removed from the event. When the `Trimmer` receives an event without a ^uT, it maintains the relevant attributes according to all categories, meaning if an attribute belongs to a group with the relevancy above the defined threshold in at least one unified taxonomy category, that attribute will not be removed from the event. In another words, the `Trimmer` processes an event with no ^uT the same way as that event had all Tier 1 ^uT.

Figure 4.4 shows the transformation of an event $E_{a'}$ processed by the `Trimmer`. In order to preserve the original event $E_{a'}$ (`Classifier` output), a new event E_b is created with the same information of $E_{a'}$, with the exception of the attributes (and their relations) and non unified taxonomy tags. When $E_{a'}$ is processed, attributes A_1 to A_m are scanned and "classified" according to the attribute groups. Based on a defined relevancy threshold `t`, for each attribute A_x, if A_x relevancy is greater or equal to `t`, A_x is added to E_b. Finally, for each relation in $E_{a'}$, if both attributes that constitute the relation were added to E_b, the relation is added to E_b.

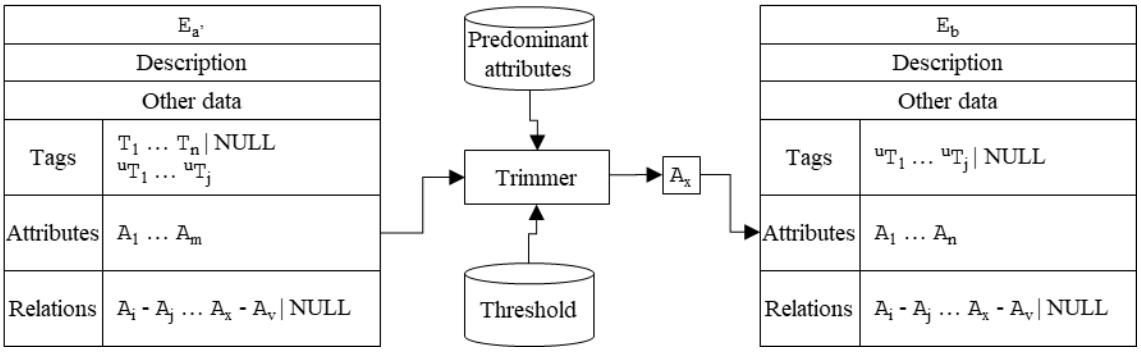

Figure 4.4: Representation of an event processed by the Trimmer

4.5 OSINT-based event enrichment

As explained in Chapter 3, Section 3.4, more than 90% of the links pointed to VirusTotal online platform. The references to external platforms increases the time that an analyst requires to analyse an event since the analyst needs to manually jump between platforms to gather information. Moreover, enriching events with additional information gathered from external platforms can significantly improve other processes and tasks, if the obtained information is related to a predominant attribute group. In order to take advantage of the references to external platforms, and so enrich the threat intelligence quality, we propose an event `Enricher`.

The `Enricher` automatically enriches events that contain attributes with links to VirusTotal, URLs or file hashes. Each event served as an input to the `Enricher` will have its attributes scanned. Each scanned attribute is parsed to extract the URLs and file hashes. Since VirusTotal links contain the IoCs in the target URL, the previous step also applies to them. For each extracted IoC (URL or file hash), a request is sent to VirusTotal, and as response is received a report containing a summary of the output of the most known antivirus engines, website scanners and analysis tools regarding that IoC. Additionally, complementary information can be received like hashes according to different hashing algorithms. This complementary information updates the source attribute transforming it in an enriched attribute. (^eA_x) Moreover, a new associated enriched attribute ($^eA_{x.1}$) to support the output of antivirus engines, website scanners and analysis tool is created, added to the event and related to the enriched attribute that was updated with the complementary information (relating the pair ^eA_x - $^eA_{x.1}$).

Figure 4.5 shows the transformation of an event E_b processed by the `Enricher`. When E_b is processed, attributes A_1 to A_n are scanned to identify and extract URLs and file hashes. Being A_x an attribute with an URL or hash, a request to VirusTotal public API is sent containing the extracted IoC from A_x. Based on the information in the response, A_x is updated to an enriched attribute ^eA_x and added to $E_{b'}$. Furthermore, an additional enriched attribute $^eA_{x.1}$ containing the output of antivirus and similar tools is created, added to $E_{b'}$ and related to ^eA_x. Finally, all the attributes that did not had an URL or file hashes are added to $E_{b'}$. In summary the result of processing of Enricher is E_b with some or all of its A_x enriched (denoted as ^eA_x) and some new $A_{x.1}$ resulting in $E_{b'}$.

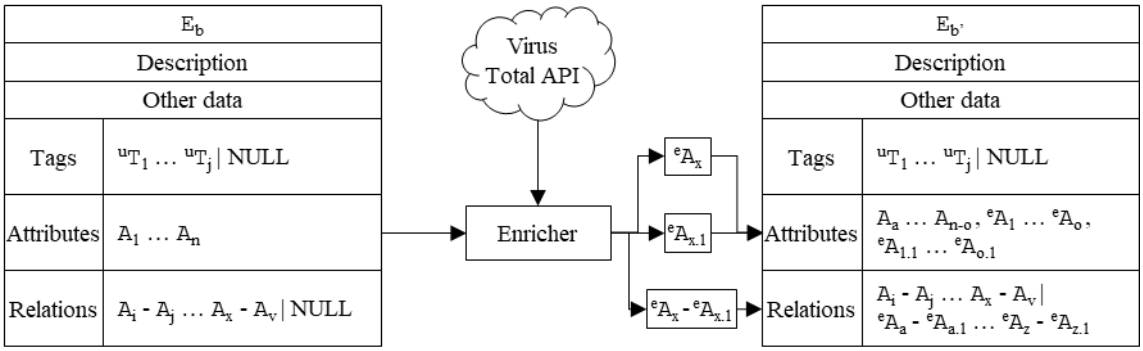

Figure 4.5: Representation of an event processed by the Enricher

4.6 Event clustering

The ability to create correlations between events is one key feature that helps threat analysts identifying threats with similarities, such as source, target, payload, threat actor and used tools. However, as mentioned in Section 2.4.3, most TIPs have limited advanced analytics capabilities related to event correlation. MISP has its own built-in correlation algorithm that allows an analyst to identify events that have attributes in common. However, this algorithm relies in the values of the attributes and one key information, a flag, that specifies if that attribute can be correlated. This flag is inserted manually and, if not used properly, have a negative impact in the correlation of events. For example, if a user adds an attribute to an event that indicates that the payload was sent over HTTP, the correlation of this attribute with attributes from other events will mostly be useless, since many attacks use HTTP to send payload. This is why some attributes should not be flagged as correlation information. Thus, it is important to manage event correlation properly. Moreover, this built-in algorithm does not use the information related to the category of the event, creating relation between events without context. In order to improve the event correlation capabilities, we propose an event `Clusterer`.

The `Clusterer` automatically creates clusters of events that share the same category and have at least one valuable attribute in common (attributes that provide context to a specific attack, e.g., hashes). Each event served as an input to the `Clusterer` will have its attributes scanned. For each scanned attribute, if it does not add value when correlated, the attribute is skipped. For example, booleans, dates and small sets of possible values like http-methods, do not add value since multiple events with no relation have them in common. Using a more concrete example, an HTTP flood attack is categorized according our unified taxonomy as `unified:availability="dos-or-ddos"` and an intrusion using an unknown exploit as `unified:intrusion-or-attempts="unknown-exploit`, both of these events can be exploiting HTTP GET method without any correlation. If the scanned attribute adds values when correlated, a search is made over the set of events to identify other events that contain the same attribute. If at least one event as a correlation with the original event and both share a unified category tag, a cluster is created. This cluster contains unified category tag ($^uT_1$:$^uT_2$) shared by all events that compose the cluster, as well as all their attributes. Finally, all events that compose the cluster are added as attributes and, for each, relations are created with the attributes that were obtained from the correspondent source events.

Figure 4.6 shows the transformation of an event $E_{b'}$ processed by the `Clusterer`. When processed, attributes A_1 to A_f are scanned to identify valuable attribute (attributes that provide context to a specific attack). Being A_x an valuable attribute, a search is made over the database to identify other events with A_x. Being $E_{c'}$ an event that contains A_x in common with $E_{b'}$, tags from $E_{b'}$ and $E_{c'}$ are scanned in order to find at least one unified category tag in common. Being ^uT_i a common tag for $E_{b'}$ and $E_{c'}$, a

new event ${}^{uT_i}Cluster_{ab}$ is created with the tag ${}^{u}T_i$. Furthermore, all the attributes from $E_{b'}$ and $E_{c'}$ are added to the cluster. Additionally, $E_{b'}$ and $E_{c'}$ are also added as attributes to represent pseudo-events. Finally, for each pseudo-event added, relations are created with the other attributes based on the original relations of the corresponding source events.

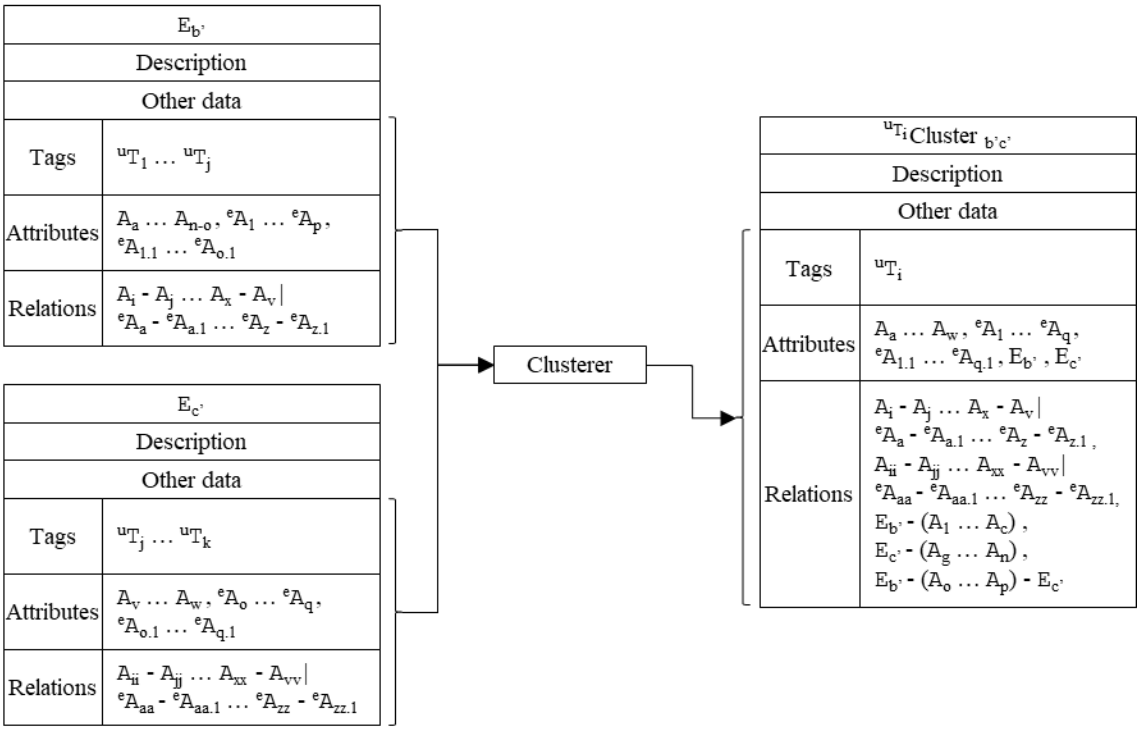

Figure 4.6: Representation of two events processed by the Clusterer

Figure 4.7 shows a more tangible example of how the `Clusterer` processes three events (E_1, E_2 and E_3) with a unified taxonomy tag in common (${}^{u}T_2$) between them. Each attribute found in common between E_1, E_2 and E_3 is added to ${}^{uT2}Cluster_{1,2,3}$ with information from each event concatenated into a single attribute. For example, A_1 is an attribute in common between E_1 and E_2 events and when added to ${}^{uT2}Cluster_{1,2,3}$ both the information from E_1 and E_2 is concatenated to form a single attribute (A_1 = [$A_{1.e1}$|| $A_{1.e2}$]), in order to not create duplicated attributes. Moreover, E_1, E_2 and E_3 are added as attributes to ${}^{uT2}Cluster_{1,2,3}$ and each concatenated attribute related to them added as a relation. For example, a relation between A_1 and E_1, and A_1 and E_1 was created.

In Chapter 6, Section 4 a real example is provided to better understand the `Clusterer` output.

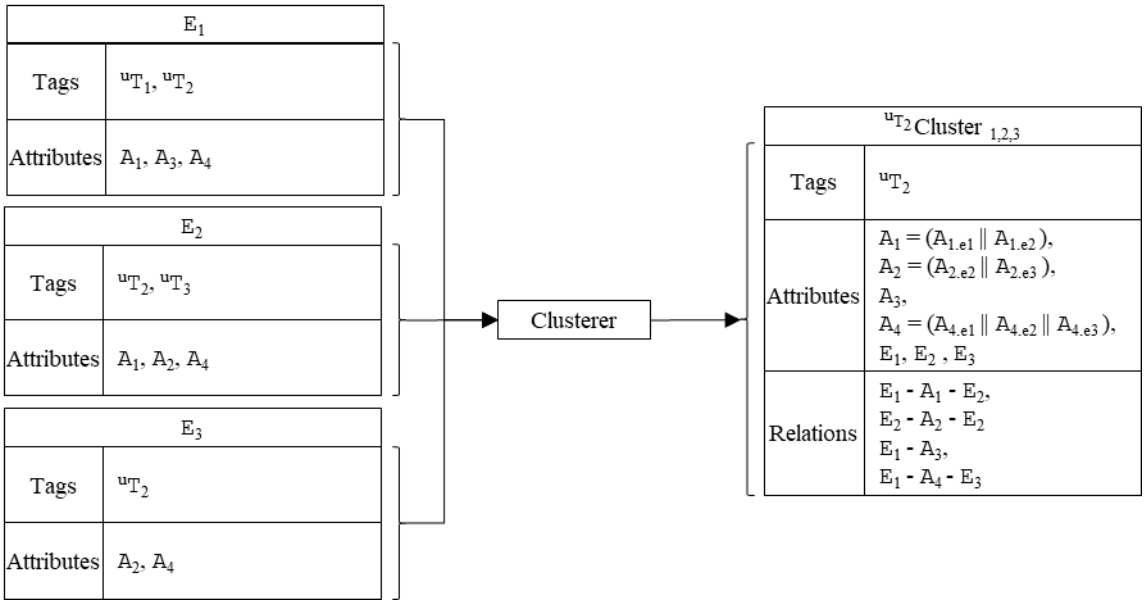

Figure 4.7: Tangible representation of three events processed by the Clusterer

Chapter 5

Implementation

In this chapter we present the high-level implementation of AECCP, following the architecture defined for each one of the modules of our platform (`Classifier`, `Trimmer`, `Enricher` and `Clusterer`), in Chapter 4. AECCP was implemented using Python 3.7 and PyMISP [39], a Python library to access MISP platforms via their REST API. The implementation of AECCP leverages from built-in PyMISP functionalities to search, add or update events and attributes, some of them mentioned in the following sections according to their use. Sections 5.1-5.4 details the implementation of each module, respectively, `Classifier`, `Trimmer`, `Enricher`, and `Clusterer`.

5.1 Classifier

As previously shown in Chapter 4, Section 3, the `Classifier` is capable of processing events without any classification tags, events not yet classified with the Unified Taxonomy (in other words, without Tier 1 nor Tier 2 tags of the Unified Taxonomy), and events only classified with Tier 1 tags of the Unified Taxonomy. Algorithm 5.1 represents the main logic behind the `Classifier`, where the processing of each event is separated in Tier 1 classification and Tier 2 classification based on the state of the event that was passed into the `Classifier`.

```
1  Function Classify
2     Pass In: event
3     IF event is not classified with UT
4        event is Ea
5        call: ClassifyTier1
6     ELSE IF event is classified with tier 1 UT
7        event is Ea'
8        call: ClassifyTier2
9     ENDIF
10 Endfunction
```

Algorithm 5.1: Algorithm of the Classifier implementation – main logic

Events not yet classified with the Unified Taxonomy, including events without any classification tags, follow the Tier 1 classification of the `Classifier` (lines 3-5). In contrast, events already classified with a Tier 1 Unified Taxonomy, follow the Tier 2 classification of the `Classifier` (lines 6-8). The implementation of the Tier 1 classification uses the public taxonomy mapping and the bag of words explained in Chapter 3, Section 3.2 and defined in Table 5.1 and Table 5.4. Algorithm 5.2 represents the logic behind the Tier 1 classification of the `Classifier`.

```
1  Function ClassifyTier1
2     Pass In: Ea
3     FOR each tier1 UT in the public taxonomy mapping
4        FOR each public taxonomy related to the tier1 UT
5           IF Ea has public taxonomy
6              Add tier1 UT tag to Ea tag list
```

```
        ENDIF
      ENDFOR
    END FOR
    FOR each tier1 UT in the bag of word
      FOR each word related to the tier1 UT
        IF Ea contains word in its description
          Add tier1 UT tag to Ea tag list if not already
        ELSE
          FOR each attribute in Ea
            IF attribute contains word
              Add tier1 UT tag to Ea tag list if not already
            ENDIF
          ENDFOR
        ENDIF
      ENDFOR
    ENDFOR
Endfunction
```

Algorithm 5.2: Algorithm of the Classifier implementation – Tier 1 classification

Similar to the implementation of the Tier 1 classification, the implementation of the Tier 2 classification also uses the public taxonomy mapping and the bag of words explained in Chapter 3, Section 3.2 and defined in Table 5.2 and Table 5.4. Algorithm 5.3 shows the logic behind the Tier 2 classification of the `Classifier`.

```
Function ClassifyTier2
  Pass In: Eb'
  Get Ea'
  FOR each tier1 UT tag in Ea' tag list
    FOR each tier2 UT related to the tier1 UT in the public taxonomy mapping
      FOR each public taxonomy related to the tier2 UT
        IF Ea' has public taxonomy
          Add tier1:tier2 UT tag to Eb' tag list if not already
        ENDIF
      ENDFOR
    ENDFOR
    FOR each tier2 UT related to the tier1 UT in the bag of word
      FOR each word related to the tier2 UT
        IF Eb' contains word in its description
          Add tier1:tier2 UT tag to Eb' tag list if not already
        Else
          FOR each attribute in Eb'
            IF attribute contains word
              Add tier1:tier2 UT tag to Eb' tag list if not already
            ENDIF
          ENDFOR
        ENDIF
      ENDFOR
    ENDFOR
  ENDFOR
  FOR each tier1 UT tag in Eb' tag list
    remove tier1 UT tag
  ENDFOR
Endfunction
```

Algorithm 5.3: Algorithm of the Classifier implementation – Tier 2 classification

5.2 Trimmer

As explained in Chapter 4, Section 4, the main function of the `Trimmer` is to reduce the quantity of not so useful information of each event, the "good to know" information, while preserving the "need to know" information, the most useful information for their analysis. The `Trimmer` is capable of processing events that passed through the Tier 1 classification of the `Classifier` (i.e., events that already contain Tier 1 tags of the Unified Taxonomy). However, some of the events that pass through the Tier 1 classification are not classified according to the Unified Taxonomy, due to the lack of information in the event. For these events, the `Trimmer` handles them the same way as they contained every single one Tier1 classification of the Unified Taxonomy in order to not lose any "need to know" information. Algorithm 5.4 shows the logic behind the `Trimmer`, which follows the process discussed previously in the section mentioned above.

```
1  Function Trim
2    Pass In: Ea'
3    Pass In: threshold
4    Create Eb as a copy of Ea'
5    Remove attributes from Eb
6    FOR each non UT tag in Eb tag list
7      Remove tag from Eb tag list
8    ENDFOR
9    Create PG list
10   IF Eb tag list is empty
11     FOR each UT in the public taxonomy mapping
12       IF attribute group predominancy percentage is hight than the threshold
13         Add attribute group to PG list if not already
14       ENDIF
15     ENDFOR
16   ELSE
17     FOR each UT tag in Eb tag list
18       FOR each attribute group in the predomiant attribute list related to UT
19         IF attribute group predominancy percentage is hight than the threshold
20           Add attribute group to PG list if not already
21         ENDIF
22       ENDFOR
23     ENDFOR
24   ENDIF
25   FOR each attribute in Ea'
26     FOR each attribute group in PG list
27       IF attribute type is related to attribute group
28         Add attribute to Eb
29       ENDIF
30     ENDFOR
31   ENDFOR
32   FOR each attribute relation in Ea'
34     IF both attributes in Eb
35       Add attribute relation to Eb
36     ENDIF
37   ENDFOR
38 Endfunction
```

Algorithm 5.4: Algorithm of the Trimmer implementation

5.3 Enricher

The `Enricher` enriches attributes that contain a file hash or an url by collecting OSINT from a known valid source and adding it to the event. As explained in Chapter 4, Section 5, we chose VirusTotal as our external source of OSINT to enrich attributes because, in the initial dataset, 90% of the attributes of the type link pointed to VirusTotal. The `Enricher` processes events that passed through the `Trimmer`, in other words, events that already contain Tier 1 tags of the Unified Taxonomy and only have “need to know” attributes. Algorithm 5.5 illustrates the main logic behind the `Enricher`, which follows the process presented previously.

```
1  Function Enrich
2    Pass In: Eb
3    FOR each attribute in Eb
4      IF attribute type is in File hash attribute group or if attribute type is url or link
5        Get data related to attribute from VirusTotal
6        Add data to attribute
7        Create AV summary attribute
8        Add AV summary attribute to Eb
9        Relate both attributes
10     ENDIF
11   ENDFOR
12 Endfunction
```

Algorithm 5.5: Algorithm of the Enricher implementation

5.4 Clusterer

As described in Chapter 4, Section 6, the `Clusterer` automatically creates clusters of events that share the same category and have at least one valuable attribute in common. The `Clusterer` only processes events that passed through every other module (`Classifier`, `Trimmer` and `Enricher`), i.e., events that were trimmed, enriched and classified with Tier 1 and Tier 2 tags of the Unified Taxonomy. The `Clusterer` search recursively upon each event and the events that share at least one attribute with that event. Algorithm 5.6 depicts the logic behind the recursive search of the `Clusterer`. The recursive search takes advantage of a MISP build-in function to get other events with at least one attribute in common with a particular event.

```
1  Function ClusterAux
2    Pass In: event
3    Pass In: UT tag
4    Get other events with at least one attribute in common with event
5    FOR each other event already trimmed, enriched and classified
6      IF other event has the UT tag passed in
7        IF attribute in common is a valuable attribute
8          add event to cluster's event list if not already
9          call: ClusterAux
10       ENDIF
11     ENDIF
12   ENDFOR
13 Endfunction
```

Algorithm 5.6: Algorithm of the Clusterer implementation – recursive search

After gathering all the events that share a specific UT tag and have at least one valuable attribute in common, the `Clusterer` adds these events to the created cluster. Each cluster was implemented as a MISP event with some special attributes that represent the events gathered and integrated in the cluster. Moreover, the `Clusterer` deduplicates the attributes in common and creates attribute relations with all the attributes and the related “events” (special attributes). Algorithm 5.7 shows the remaining logic behind the `Clusterer`, which implements these steps.

```
1  Function Cluster
2    Pass In: Eb'
     FOR each UT tag in event tag list
3      Create cluster
4      call: ClusterAux
5      FOR each event in cluster's event list
6        Create an "event" attribute with the event id as value
7        FOR each attribute in event
8          Add attribute to cluster if not already
9          Create attribute relation between the attribute and the "event" attribute
10       ENDFOR
11       FOR each attribute relation in event
12         Add attribute relation to cluster
13       ENDFOR
14     ENDFOR
     ENDFOR
15 Endfunction
```

Algorithm 5.7: Algorithm of the Clusterer implementation – main logic

5.5 Orchestrator

In the previous sections of Chapter 5, we explained how each core module of our platform was implemented. These modules were implemented to work independently of each other and can be used in a custom order, if one desires. However, to achieve the best possible results each event requires to follow a specific flow through each module, as explained in Section 4.2. To achieve this, we implemented an `Orchestrator`. This module is responsible to assure that each event, at any time, follows a specific flow and it is only processed by a module if the event has the required reequipments (e.g., only can be enriched if it was already trimmed). Additionally, this module is responsible to check for new events that were added to our MISP instance (via sharing or manual creation), and to initiate the AECCP processing for each event. In sum, the `Orchestrator` is responsible to periodically fetch and initiate the AECCP processing for new events, assure the correct processing order for each event, and resume the processing of events if they are interrupted before completing the full AECCP process.

- **Periodically fetch new events:** The `Orchestrator` periodically checks if there are new events from the selected feeds and adds them to our MISP instance, leveraging from PyMISP built in methods.
- **Initiate processing of new events:** The `Orchestrator` periodically checks for events that were added since last time AECCP processed an event.
- **Assure the correct processing order:** The `Orchestrator` acts as a manager by sending each event to the correct next module. This module take advantage of custom tags that are only used by the `Orchestrator`. These tags stores the current state of the event regarding AECCP processing order.
- **Resume the process:** If the process of an event is interrupted, the `Orchestrator` is able to resume the processing of that event without impacting the event database by falling back to the previous event state.

Chapter 6

Evaluation

In this chapter we present the evaluation of AECCP platform. This evaluation aims at validating AECCP ability to enrich, classify and correlate events. In another words, this chapter presents the evaluation performed over the implementations of the `Classifier`, `Trimmer`, `Enricher` and `Clusterer` modules described in Sections 5.1 to 5.4. Also, in this chapter, a description of the used data sources is provided, as well as the dataset used in the evaluation. More specifically, we looked to answer the following questions:

1. Is AECCP able to classify events that are not initially classified?
2. Is AECCP able to reclassify events previously classified with a known incident classification taxonomy?
3. Does AECCP simplifies event triage?
4. Is Trimmer able to reduce the number of attributes of events without losing valuable information for their classification?
5. Does Enricher improve the quality of the events?
6. Is AECCP able to correlate different events (threats) that share the same IoC?

6.1 Data characterization

To evaluate our solution, we followed a similar approach as the one used in Chapter 3, Section 3.1, but with events from a different time period that were not used in the initial analysis dataset, i.e., the one used in Section 3.1. Firstly, we will show the distribution of the events from evaluation dataset according to their providers. Secondly, the dataset will be characterized according to the initial classification of its events, and next to the volume of attributes per event.

The events from March 1st, 2019 until July 31th, 2019 from the three MISP formatted feeds used in the data analysis (CIRCL OSINT Feed, The Botvrij.eu Data and inThreat OSINT Feed) formed the dataset used to evaluate our solution. In total, the evaluation dataset contained 64 events. Figure 6.1 shows the distribution of the events of the evaluation dataset according to their provider. Providers with less than or equal to 5 events were aggregated into “Other”, including, but not exclusively, VK-Intel, ESET and MalwareMustDie.

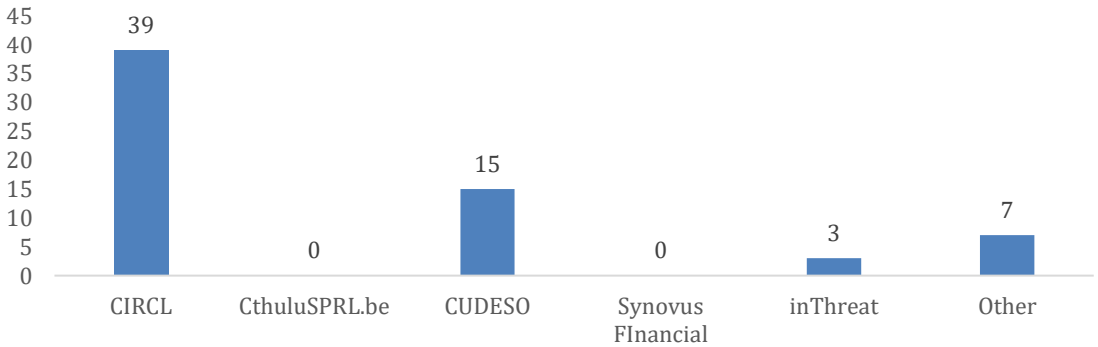

Figure 6.1: Distribution of events by provider started from March 2019 until July 2019

Regarding the event classification our evaluation dataset, from the 64 events that formed it, approximately 77% (49) of them did not contain any tags related to a known incident classification taxonomy, meaning that those events were not classified. This information will be used to evaluate the AECCP ability to classify events with the *classification based on keywords*, detailed in Section 4.3 and evaluated in Section 6.2. Figure 6.3 shows the number of events from the dataset initially classified with a public taxonomy.

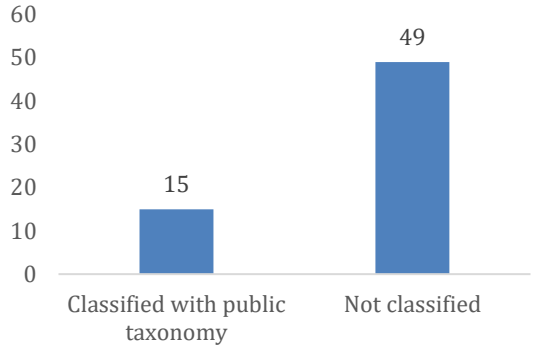

Figure 6.2: Events from the evaluation dataset initially classified with a public taxonomy

Regarding the volume of attributes of the events, our dataset was mainly composed of events with less than 100 attributes, approximately 91% of the 64 events. Figure 6.3 shows the distributed per number of attributes according to the same four intervals used in Chapter 3, Section 3.3 (]0,100],]100,500],]500,1000] and]1000,+∞[).

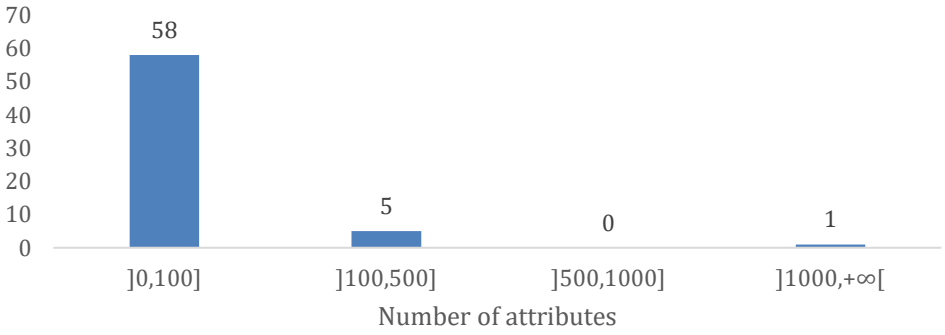

Figure 6.3: Events from the evaluation dataset per number of attributes

In order to get a detailed evaluation of our solution, we choose to perform a more in-depth analysis of the 15 events that were initially classified with a known incident classification taxonomy. We choose these events since they can be used to evaluate almost all use cases, except AECCP ability to classify events that are not initially classified, which can be evaluated comparing the number of unclassified events initially and after being processed by AECCP. Table 6.1 shows a more detailed view of the tags and the attributes of the 15 events from our evaluation dataset that were initially classified with a known incident classification taxonomy. More specifically, Table 6.1 shows their public taxonomy tags, the total number of tags, including tags that did not add information about the type of the threat, such as the Traffic Light Protocol, and the number of attributes.

Table 6.1: Tags and attribute details from 15 events from the evaluation dataset

Event	Public taxonomies tags	Total no. of tags	No. of attributes
1	circl:incident-classification="spam"	12	17
2	enisa:nefarious-activity-abuse="spear-phishing-attacks"	4	84
3	malware_classification:malware-category="Botnet"	4	10
4	malware_classification:malware-category="Ransomware"	5	18
5	malware_classification:malware-category="Ransomware"	3	9
6	circl:incident-classification="malware" malware_classification:malware-category="Downloader" malware_classification:malware-category="Rootkit" malware_classification:malware-category="Botnet"	8	73
7	malware_classification:malware-category="Ransomware"	5	7
8	circl:incident-classification="malware"	8	29
9	circl:incident-classification="malware"	4	11
10	enisa:nefarious-activity-abuse="spear-phishing-attacks"	8	115
11	ecsirt:intrusions="backdoor" veris:action:malware:variety="Backdoor" ms-caro-malware:malware-type="Backdoor" ms-caro-malware-full:malware-type="Backdoor"	38	17
12	ms-caro-malware:malware-type="Trojan" ms-caro-malware-full:malware-type="Trojan" ecsirt:malicious-code="trojan" cert-xlm:malicious-code="trojan-malware" malware_classification:malware-category="Trojan"	10	10
13	ecsirt:intrusions="backdoor" veris:action:malware:variety="Backdoor" ms-caro-malware:malware-type="Backdoor" ms-caro-malware-full:malware-type="Backdoor"	10	34
14	circl:incident-classification="malware" ecsirt:malicious-code="malware"	12	86
15	ecsirt:malicious-code="trojan"	7	27

6.2 Event classification

In this section we look to evaluate AECCP ability to classify events. More precisely, we will evaluate the `Classifier` implementation, the main module of this functionality, but also the `Trimmer` and `Enricher` implementations since both of these modules support the `Classifier` in the classification of events. During this section we will compare events in E_a state with events in $E_{b'}$ state, excluding the `Clusterer` which will be analysed separately in Section 6.4. Figure 6.4 shows the main module that contributes to the event classification, the `Classifier`, and the modules supporting it, the `Trimmer` and the `Enricher`.

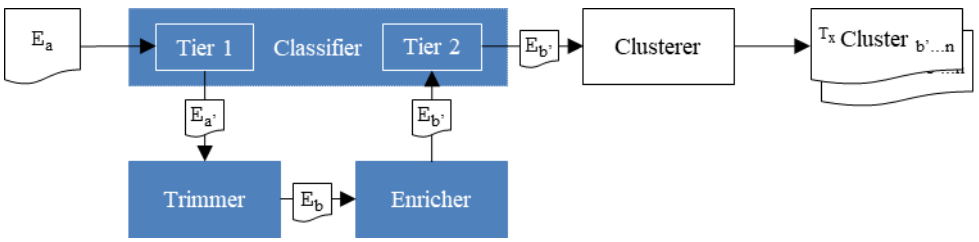

Figure 6.4: Modules that contribute to event classification

This section will seek the answer to the first three questions we defined, namely:

1. Is AECCP able to classify events that are not initially classified?
2. Is AECCP able to reclassify events previously classified with a known incident classification taxonomy?
3. Does AECCP simplifies event triage?

Before being processed by AECCP, our dataset contained 49 of the 64 events without any tags related to a known incident classification taxonomy. After being processed by AECCP, only three events were not classified according to the Unified Taxonomy, due to the lack of information in their descriptions and the absence of indicators that could be processed by the `Enricher`, such as URL and file hashes, thus adding more information to the events. This results in a 72% increase of the number of classified events and a total of 61 classified events from 64 events. Moreover, if we subtract the 15 events that were initially classified from these results, we obtain the number of events that were classified by the `Classifier` only using the *classification based on keywords*, since there were no tags to use on the *classification based on public taxonomies tags*. 75% (46) of 61 classified events by AECCP were classified only based on keywords, meaning that AECCP is able to classify events that are not initially classified, answering question 1. Figure 6.5 compares the number of events from the evaluation dataset classified before and after being processed by AECCP.

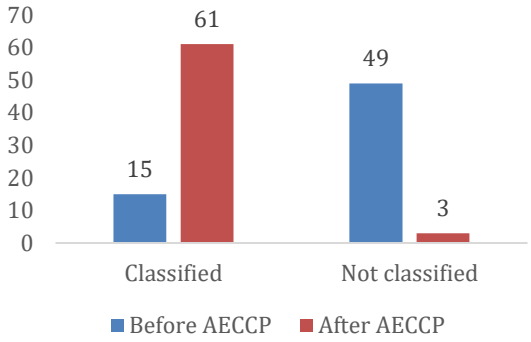

Figure 6.5: Comparison of event classification before and after being processed by AECCP

Regarding the analysis targeted to the 15 events that were initially classified with a known incident classification taxonomy, the platform was able to use both methods (*classification based on keywords* and *classification based on public taxonomies tags*) since these events had public taxonomy tags in their tag lists. Almost every event of these 15 events was classified with a new type of threat that was not initially considered in the public taxonomy tags. For example, event #1 from Table 6.1 was identified only as spam before being processed by AECCP. However, after being processed by AECCP, it was also identified as malicious code with virus, worm and spammer behaviours, meaning that AECCP is able to reclassify events previously classified with a known incident classification taxonomy, answering question 2. Table 6.2 shows the transformation of the tags of the 15 events that were initially classified with a known incident classification taxonomy.

Table 6.2: Reclassification of the 15 events to the Unified Taxonomy by AECCP

Event	Public taxonomies tags (before AECCP)	Unified taxonomy tags
1	circl:incident-classification="spam"	malicious-code="virus" malicious-code="worm" malicious-code="spammer" abusive-content="spam"
2	enisa:nefarious-activity-abuse="spear-phishing-attacks"	fraud="phishing"
3	malware_classification:malware-category="Botnet"	availability="dos-or-ddos" malicious-code="exploit" malicious-code="dos" malicious-code="backdoor" malicious-code="remote-access-tool" malicious-code="cryptominer"
4	malware_classification:malware-category="Ransomware"	vulnerable="vulnerable-service" malicious-code="exploit" malicious-code="ransomware"
5	malware_classification:malware-category="Ransomware"	malicious-code="wiper" malicious-code="ransomware"
6	circl:incident-classification="malware" malware_classification:malware-category="Downloader" malware_classification:malware-category="Rootkit" malware_classification:malware-category="Botnet"	malicious-code="virtool" malicious-code="cryptominer" malicious-code="trojan" malicious-code="remote-access-tool"
7	malware_classification:malware-category="Ransomware"	malicious-code="ransomware"
8	circl:incident-classification="malware"	malicious-code="virus" malicious-code="trojan"
9	circl:incident-classification="malware"	malicious-code="trojan"
10	enisa:nefarious-activity-abuse="spear-phishing-attacks"	fraud="phishing"
11	ecsirt:intrusions="backdoor" veris:action:malware:variety="Backdoor" ms-caro-malware:malware-type="Backdoor" ms-caro-malware-full:malware-type="Backdoor"	malicious-code="virtool" malicious-code="trojan" malicious-code="backdoor" fraud="phishing"
12	ms-caro-malware:malware-type="Trojan" ms-caro-malware-full:malware-type="Trojan" ecsirt:malicious-code="trojan" cert-xlm:malicious-code="trojan-malware" malware_classification:malware-category="Trojan"	malicious-code="trojan"
13	ecsirt:intrusions="backdoor" veris:action:malware:variety="Backdoor" ms-caro-malware:malware-type="Backdoor" ms-caro-malware-full:malware-type="Backdoor"	malicious-code="virtool" malicious-code="backdoor" malicious-code="virus" malicious-code="cryptominer"
14	circl:incident-classification="malware" ecsirt:malicious-code="malware"	malicious-code="trojan"
15	ecsirt:malicious-code="trojan"	malicious-code="trojan"

As explained in Sections 3.2 and 4.3, AECCP classifies events according to the Unified Taxonomy in order to eliminate overlapping classification tags. In addition, AECCP also classifies events based on information contained in their description, meaning that each event classification can be improved. This results in an increase of the number of tags per event. On average, each event had more 5 tags than before being processed by AECCP, increasing their tags from 2 to 7. It is important to note that, after being processed by AECCP, all of the tags on the events tag list are classification tag, in contrary to before being processed by AECCP where most tags were not classification tags, but added information about its source and its sharing, such as the Traffic Light Protocol (TLP). Table 6.3 shows the number of tags of the 15 events that were initially classified with a known incident classification taxonomy, before and after being processed by AECCP.

Regarding the AECCP impact on the 15 events initially classified by MISP, 14 of them had their total number of tags significantly reduced (columns 2 and 4). The number of total tags can be reduced due two factors. The first is when an event has overlapping classification tags in its initial tag list (i.e., `cert-xlm:malicious-code="ransomware"` and `cccs:malware-category="ransomware"`) since they are transformed into a single unified taxonomy tag after

being processed by AECCP. The second one is when an event has non-classification tags in its initial tag list (i.e., TLP) since they are removed after being processed by AECCP. However, the number of total tags can increase if the number of newly added classification tags is higher than the number of removed tags. Additionally, half of theses 15 events had their number of classification tags increased (columns 3 and 5). The number of classification tags can increase, decrease or maintain depending on the initial number of overlapping classification tags and the number of newly added classification tags.

From the point of view of a SOC analyst the exclusion of non-classification tags and the inclusion of new classification tags based on OSINT can simplify event triage since all the tags in the event tag list add value to the analyses, answering question 3.

Table 6.3: Number of tags from 15 events before and after being processed by AECCP

Event	Before AECCP		After AECCP	
	Total tags	Classification tags	Total tags	Classification tags
1	12	1	4	4
2	4	1	1	1
3	4	1	6	6
4	5	1	3	3
5	3	1	2	2
6	8	4	4	4
7	5	1	1	1
8	8	1	2	2
9	4	1	1	1
10	8	1	1	1
11	38	4	4	4
12	10	5	1	1
13	10	4	4	4
14	12	2	1	1
15	7	1	1	1

6.3 Attribute trimming and enrichment

In this section we look to evaluate AECCP ability to trim and enrich events. More precisely, we will evaluate the `Trimmer` and `Enricher`, the modules that have these functionalities. During this section we will compare events in E_a state with events in E_b and $E_{b'}$ state. Figure 6.6 shows the main modules that contributes to the event trimming and enrichement, the `Trimmer` and the `Enricher`.

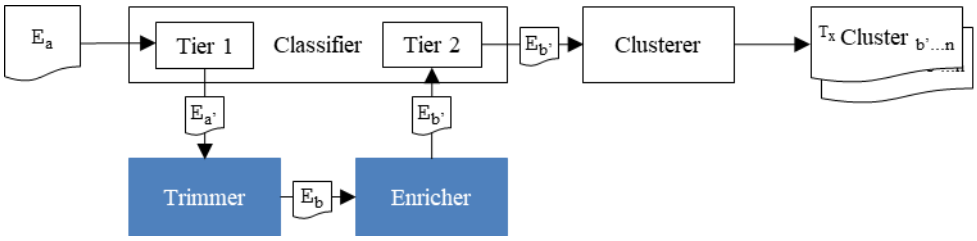

Figure 6.6: Modules that contribute to event trimming and enrichment

This section will seek the answer to the fourth and fifth questions defined previously:

4. Is `Trimmer` able to reduce the number of attributes of events without losing valuable information for their classification?
5. Does `Enricher` improve the quality of the events?

Before being processed by AECCP, our dataset had, approximately, 90% of the events with less than 100 attributes. After being processed by AECCP, the number of events with less than 100 attributes decreased to 85% of the initial number. This means that our solution enriches more than it trims, adding more attributes than removing. Figure 6.7 compares the number of attributes of the events from the evaluation dataset before and after being processed by AECCP.

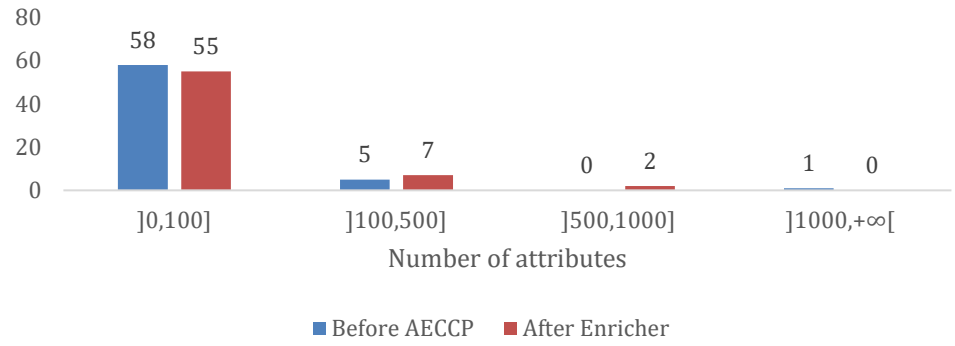

Figure 6.7: Events from the evaluation dataset per number of attributes before and after being processed by AECCP

In order to understand the overall increase in the number of attributes per event after being processed by AECCP, we analysed the number of attributes of the events in three specific phases: before being processed by the `Trimmer`, exactly after being processed by the `Trimmer` and, finally, after being processed by the `Enricher`. From the results of this analysis, we can see that, on average, the trimmer removes 12 attributes per event and the enricher adds 54 attributes per event, resulting in a increase of 44 attributes per event. This increasing made by the Enricher is because it can add a maximum of, for each hash, 6 new attributes and, for each URL, 12 new attributes. Therefore, if an event has 3 hashes and 3 URLs, the `Enricher` will add 54 attributes to the event. Figure 6.8 compares the average number of attributes per event before being processed by AECCP, exactly after being processed by the `Trimmer` and the `Enricher`.

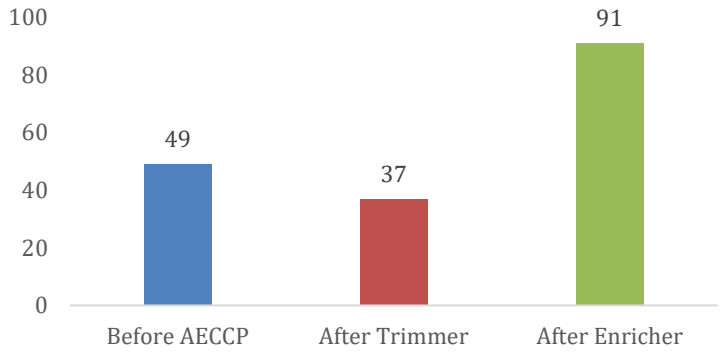

Figure 6.8: Average number of attributes per event before and while being processed by AECCP

Similar to the `Classifier` evaluation, in Section 6.2, we also evaluated the `Trimmer` and `Enricher` impact on the 15 events that were initially classified with a known incident classification taxonomy. From these 15 events, 6 had the number their attributes increased, 3 had the number their attributes reduced and 6 maintained the number of their attributes. These results are aligned with Figures 6.7 and 6.8 results, showing an overall increase of the number of attributes per events. However, AECCP can reduce the number of attributes of some events depending of the type of attributes on those events. Table 6.4 shows the number of attributes from 15 events before and after being processed by AECCP.

Table 6.4: Number of attributes from 15 events before and after being processed by AECCP

Event	Before AECCP	After Trimmer	After Enricher
1	17	13	13
2	84	78	92
3	10	10	10
4	18	18	42
5	9	8	8
6	73	43	53
7	7	7	7
8	29	29	36
9	11	11	11
10	115	105	173
11	17	15	34
12	10	10	10
13	34	34	34
14	86	86	86
15	27	27	166

In order to answer if the `Trimmer` do not remove valuable information for the classification of events and if the enricher improves their quality, we made an evaluation with and without these two modules. Table 6.5 shows the results of this evaluation. The table compares the number of classification tags of the 15 events processed by AECCP if they did not pass through the Trimmer and the Enricher (column 2), with the number of classification tags if they only did not pass through the Enricher (column 3) and with the number of classification tags when processed by AECCP with all modules.

Table 6.5: Trimmer and Enricher impact on the number of tags of the 15 events

Event	Without Trimmer and Enricher	With Trimmer	With Trimmer and Enricher
1	4	4	4
2	1	1	1
3	5	5	6
4	3	3	3
5	2	2	2
6	4	4	4
7	1	1	1
8	1	1	2
9	0	0	1
10	1	1	1
11	4	4	4
12	1	1	1
13	4	4	4
14	1	1	1
15	0	0	1

As we can observe from Table 6.5 all the events have the same number of tags in columns 2 and 3 meaning that the `Trimmer` do not remove valuable information for the classification of events, answering question 4. We can also observe from column 4 that the `Enricher` increased the number of tags of three events. This is a minor improvement of the overall number of tags; however, these 15 events were already initially classified therefore harder to add new tags. Nevertheless, the `Enricher` improved the quality of the events, answering question 5.

6.4 Clustering

In this section we look to evaluate AECCP ability to correlate different events that share mutual IoCs. More precisely, we will evaluate the `Clusterer` implementation, the module of responsible for this functionality. During this section we will analyse the clusters produced in $^{Tx}Cluster_{b',...n}$ state. Figure 6.9 shows the main module that contributes to the event clustering, the `Clusterer`.

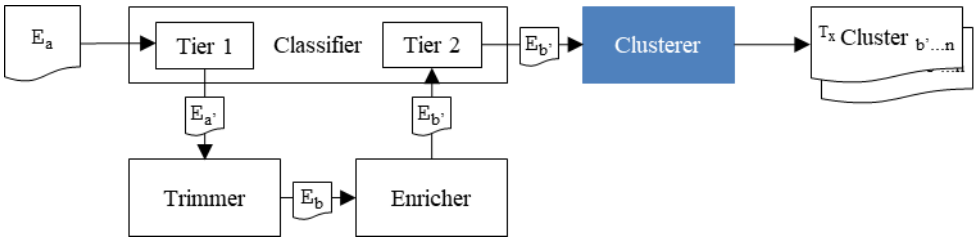

Figure 6.9: Modules that contribute to event clustering

This section will seek the answer to the following question:

6. Is AECCP able to correlate different events that share the same IoC and is that correlation helpful to a SOC analyst?

Since our evaluation dataset only contained 64 events, all from a similar date, we were not able to create many clusters. In order to overcome this problem, we allowed the events from our evaluation dataset to be correlated with events from our analysis dataset, Section 3.1. In other words, only the 64 events from our evaluation dataset were allowed to initiate the `Clusterer` module, but they could correlate with the 1168 events from the analysis dataset, resulting a total of 1232 events. With this approach we were able to create 24 clusters. Table 6.6 details some of these clusters while the remaining are omitted since they have exactly the same properties, except their taxonomies, as one of the clusters in this table. For example, clusters 100, 101 and 102 have exactly the same attributes and correlations, but they were created with different taxonomies (malicious-code="worm", malicious-code="backdoor" and malicious-code="trojan") due to the implementation of the `Clusterer` detailed in Chapter 5.4.

Figure 6.10 presents one of the clusters that were created by AECCP, identified with ID 21 in Table 6.6. This cluster is formed by two events (1518 and 1520) that have a common attribute, a link, and a common unified taxonomy tag, in this case `malicious-code="ransomware"`. The attribute in common is a link to news related to ransomware LockerGoga, meaning that both of the events are related to the same threat. Because these two events have different information, in exception to the single shared link, they complement each other. This type of event correlation can be extremely valuable to a SoC analyst, since he can easily gather more information about an event based on previously received events, giving SoC analysts more indicators that can be used in block rules and other types of defences, answering question 6. In addition to Figure 6.10, the visualization of the two event that form this cluster can be found in Appendix D.

Table 6.6: Number of tags from 15 events before and after being processed by AECCP

ID	Number of events	Taxonomy and Description	Number of Attributes	Mutual IoCs
1	2	malicious-code="worm" -Soft Cell case indicators -Malware with Ties to SunOrcal	416	www.tashdqdxp.com

4	3	malicious-code="backdloor" -Information stealer doc University Luxemburg -Malware Targeting Tibetan Diaspora Resurfaces -Multiple Cobalt Personality Disorder	451	CVE-2017-11882
7	2	malicious-code="backdloor" -Turla PowerShell blogpost -ESET Turla LightNeuron Research	75	https://www.welivesecurity.com/wp-content/uploads/2019/05/ESET-LightNeuron.pdf
9	3	malicious-code="trojan" -FIN7 JScript Loader Malware -APT28 XTunnel Backdoor -Turla Kazuar RAT	68	https://twitter.com/VK_Intel/status/1128079463785349121
10	2	malicious-code="virus" -FIN7 JScript Loader Malware -APT28 XTunnel Backdoor	47	https://twitter.com/VK_Intel/status/1128079463785349121
11	2	malicious-code="ransomware" -Sodinokibi ransomware -Ransomware exploits WebLogic vulnerability	69	All except one
14	2	malicious-code="cryptominer" -Botnet Malware Exploits CVE-2019-3396 -SystemTen (ELF trojan, miner, bot and rootkit)	65	CVE-2019-3396
16	2	malicious-code="trojan" -STUXSHOP Stuxnet Component Dials Up -Cheshire Cat	173	4e0a3498438adda8c50c3e101cfa86c5 / fa1e5eec39910a34ede1c4351ccecec8 / 7b0e7297d5157586f4075098be9efc8c / 3ba57784d7fd4302fe74beb648b28dc1
17	2	malicious-code="trojan" -Hancitor domains -Hancitor active again yith new macro	468	beetfeetlife.bit
18	2	malicious-code="trojan" -North Korean Trojan: HOPLIGHT -Malware - Gafgyt.Gen28	2361	c4103f122d27677c9db144cae1394a66
119	2	malicious-code="backdloor" -Operation ShadowHammer -Operation ShadowHammer	53	All except three
21	2	malicious-code="ransomware" -The Norsk Hydro ransomware attack -New LockerGoga Ransomware in Altran Attack	28	https://www.bleepingcomputer.com/news/security/new-lockergoga-ransomware-allegedly-used-in-altran-attack/
22	2	malicious-code="virus" -Spam Warns about Boeing 737 Max Crashes -DarkHydrus attacks targets in Middle East	47	https://twitter.com/360TIC/status/1106524508612026369
23	2	malicious-code="backdloor" -New SLUB Backdoor Uses GitHub -New SLUB Backdoor Uses GitHub	27	All except three

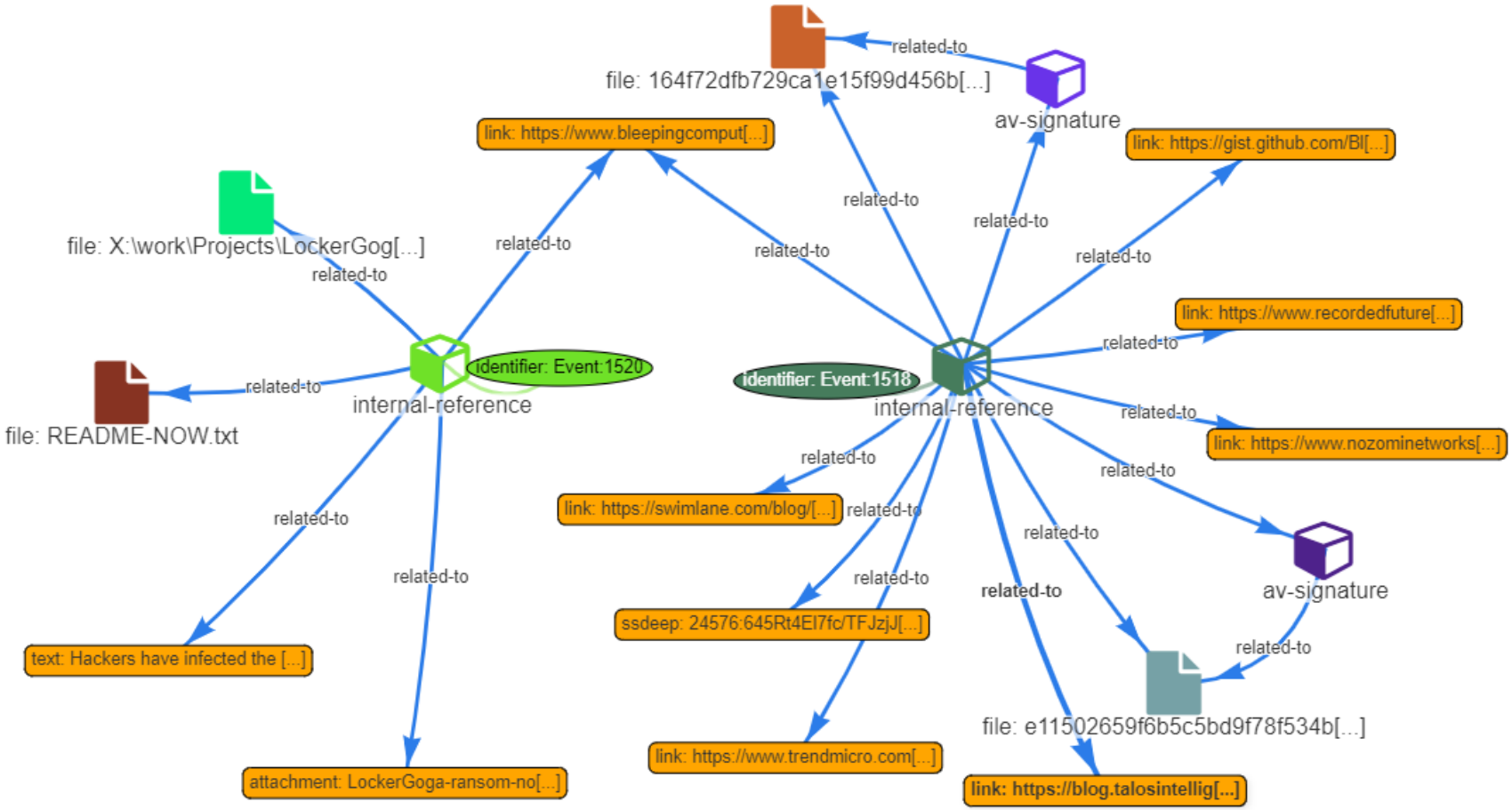

Figure 6.10: Cluster 21 created by AECCP

6.5 Processing events from other platforms

AECCP can also be used to process events processed by other platforms as long as those platforms are able to import events to a MISP instance. One example of these platform is PURE, an enriching threat intelligence platform, as an extended import, quality assessment processes and information sharing capabilities in current TIPs [32].

To demonstrate AECCP ability to process events processed by other platforms, we gathered 6 events from PURE in which their characterization is presented in Table 6.7 order to demonstrate this ability. All the 6 events were not initially classified nor timed or enriched by AECCP. Table 6.7 shows the characterization of those 6 events.

Table 6.7: Characterization of events from PURE

ID	Number of events	Description	Number of Attributes
50	2	-Expansion on 596552@qq.com -New Variant of Gh0st Malware by Palo Alto Networks Unit 42	274
51	2	-OSINT - Packrat: Seven Years of a South American Threat Actor -Packrat: Seven Years of a South American Threat Actor	267
52	4	-Sakula Malware Family -Cyber-Kraken (Threat Group 3390 / Emissary Panda) -Korean Website Installs Banking Malware -Sakula Reloaded	842
53	2	-OSINT Aveo Malware Family Targets Japanese Speaking Users -Pivot on whois registrant 844148030@qq.com	82
54	3	-EPS Processing Zero-Days Exploited by Multiple Threat Actors -Malicious Documents Targeting Security Professionals -APT28 Targets Hospitality Sector, Presents Threat to Travelers	156
55	3	-Spear Phishing Attack Using Cobalt Strike Against Financial Institutions -RTF files for Hancitor utilize exploit for CVE-2017-11882 -Targeted Attack in the Middle East by APT34, Using CVE-2017-11882	85

The 6 events received from PURE were processed by AECCP, producing the results shown in Table 6.8.

Table 6.8: PURE events processed by AECCP

ID	Number of events	Number of Attributes			AECCP Unified Taxonomy
		Before AECCP	After Trimmer	After Enricher	
50	2	274	273	401	malicious-code="backdloor" malicious-code="trojan"
51	2	267	257	423	availability="dos-or-ddos" fraud="phishing" malicious-code="backdloor" malicious-code="dos" malicious-code="ransomware" malicious-code="trojan" malicious-code="worm"

52	4	842	821	2907	information-gathering="scanning" malicious-code="backdloor" malicious-code="trojan"
53	2	82	77	87	malicious-code="backdloor" malicious-code="trojan"
54	3	156	146	361	information-gathering="scanning" malicious-code="backdloor" malicious-code="exploit" malicious-code="ransomware" malicious-code="trojan" malicious-code="worm" vulnerable="vulnerable-service"
55	3	85	78	159	abusive-content="spam" fraud="phishing" malicious-code="exploit" malicious-code="spammer" malicious-code="trojan" vulnerable="vulnerable-service"

As we can observe from Table 6.8 results, AECCP was able to process events from an external platform. All of the events, that were not initially classified, were classified by AECCP. However, as explained in Section 6.3, AECCP adds on average 44 attributes per event. This increase of attributes can be seen in Table 6.8 results, a price to pay for the added value.

Chapter 7

Conclusion

In this work we proposed and presented *Automated Event Classification and Correlation Platform* (AECCP), an implementation of an approach to improve threat intelligence quality produced by threat intelligence platforms (TIPs), by classifying and enriching it automatically. AECCP is composed of a set of smaller solutions, each one focused on one or more limitations of TIPs, which were verified in a detailed data analysis over an intelligence dataset of more than 1000 events. Regarding threat knowledge management limitations and technology enablement in threat triage limitations, we proposed a `Classifier`, a solution to classify each event according to a single unified taxonomy proposed by us. Regarding the high volume of shared threat information, we proposed a `Trimmer`, a solution that trims the low value information from each event. For data improving, we proposed an `Enricher`, a solution that enriches each event based on intelligence collected from VirusTotal. Lastly, to address advanced analytics limitations, we proposed a `Clusterer`, a solution that creates clusters of events that share information and context.

In order to prove the applicability and feasibility of AECCP, our solution was developed and implemented based around MISP, using Python 3.7 and PyMISP, a Python library to access MISP platforms via their REST API. AECCP implementation was then tested against a dataset of 64 newer and not used events from the same intelligence feed used in the initial data analysis, and 6 events produced by a different platform, PURE. Form these tests we created 24 clusters, classified, trimmed and enriched by AECCP, and we were able to trim and enrich the events produced by PURE.

7.1 Future Work

As every other work, ours presents some limitations, some derived from the novelty of this area and the time available to design, develop and implement our solution, while others were created by its design. These limitations were observed during the evaluation of our solution and can be tackled by improvements and future work developed around it.

Regarding the `Classifier`, further efforts should be placed in improving the quality of the data that supports our unified taxonomy, namely the related public taxonomies and the key words for each unified taxonomy category. This work could be further elevated to take advantage of natural language processing or other similar solution, to augment the searching and matching capabilities of the `Classifier`.

Regarding the `Trimmer`, the analysis done to obtain the most valuable information for each threat type should be done using a larger dataset from more diverse feeds to further refine the results.

Regarding the `Enricher`, the information gathered and used to enrich the events could be gathered from more platforms specific to each threat type since each threat type has different information and different requirements.

Regarding the `Clusterer`, different clustering algorithms could be applied in other to obtain faster results.

Finally, machine learning could be used to elevate each module, increasing the classification, trimming, enriching and clustering capabilities of AECCP.

Appendix A

Tag analysis results

Table A.1: Tag analysis results with more than 1 hit

Tag	Hits
tlp:white	1133
osint:source-type="blog-post"	275
Type:OSINT	273
circl:incident-classification="malware"	218
malware_classification:malware-category="Ransomware"	113
ecsirt:malicious-code="ransomware"	98
misp-galaxy:ransomware="Locky"	70
inthreat:event-src="feed-osint"	32
osint:source-type="block-or-filter-list"	32
circl:topic="finance"	31
misp-galaxy:threat-actor="Sofacy"	26
OSINT	26
misp-galaxy:tool="Trick Bot"	24
osint:source-type="technical-report"	23
workflow:todo="expansion"	22
osint:lifetime=ephemeral	21
osint:source-type=block-or-filter-list	19
ms-caro-malware:malware-platform="AndroidOS"	18
workflow:todo="create-missing-misp-galaxy-cluster-values"	17
workflow:todo="create-missing-misp-galaxy-cluster"	16
misp-galaxy:tool="Emotet"	15
misp-galaxy:ransomware="Fake Globe Ransomware"	15
ms-caro-malware:malware-type="Ransom"	14
misp-galaxy:ransomware="Jaff"	13
veris:action:social:variety="Phishing"	13
misp-galaxy:microsoft-activity-group="STRONTIUM"	12
osint:source-type="microblog-post"	12
workflow:state="incomplete"	12
ms-caro-malware-full:malware-family="Banker"	12
Android Malware	12
misp-galaxy:threat-actor="Lazarus Group"	11
malware_classification:malware-category="Trojan"	11
estimative-language:confidence-in-analytic-judgment="moderate"	10
estimative-language:likelihood-probability="very-likely"	10
osint:source-type="pastie-website"	10
ms-caro-malware-full:malware-platform="AndroidOS"	10
osint:lifetime="perpetual"	10
workflow:todo="add-missing-misp-galaxy-cluster-values"	9
enisa:nefarious-activity-abuse="mobile-malware"	9
circl:incident-classification="system-compromise"	9
circl:incident-classification="phishing"	9
tlp:green	9
ms-caro-malware:malware-platform="Linux"	9
misp-galaxy:ransomware="CryptoMix"	8
admiralty-scale:source-reliability="b"	8
enisa:nefarious-activity-abuse="ransomware"	8

ms-caro-malware:malware-type="RemoteAccess"	8
admiralty-scale:information-credibility="2"	8
admiralty-scale:information-credibility="3"	8
ecsirt:malicious-code="malware"	8
misp-galaxy:tool="Dridex"	7
APT	7
malware_classification:malware-category="Botnet"	7
osint:certainty="50"	7
misp-galaxy:tool="Turla"	6
misp-galaxy:mitre-enterprise-attack-attack-pattern="Spearphishing Attachment - T1193"	6
misp-galaxy:banker="Panda Banker"	6
misp-galaxy:mitre-enterprise-attack-intrusion-set="APT28"	6
misp-galaxy:threat-actor="Turla Group"	6
misp-galaxy:ransomware="Bad Rabbit"	6
misp-galaxy:threat-actor="Anunak"	6
enisa:nefarious-activity-abuse="remote-access-tool"	6
circl:incident-classification="scam"	6
Threat:Sofacy/APT28	6
misp-galaxy:ransomware="WannaCry"	5
misp-galaxy:preventive-measure="Backup and Restore Process"	5
misp-galaxy:tool="Mirai"	5
workflow:state="complete"	5
veris:action:malware:variety="Ransomware"	5
misp-galaxy:tool="X-Agent"	4
misp-galaxy:tool="Shamoon"	4
misp-galaxy:tool="Wipbot"	4
misp-galaxy:tool="ETERNALBLUE"	4
misp-galaxy:tool="Hancitor"	4
misp-galaxy:preventive-measure="Restrict Workstation Communication"	4
osint:certainty="75"	4
ms-caro-malware:malware-platform="Win32"	4
admiralty-scale:information-credibility="6"	4
dnc:malware-type="CoinMiner"	4
admiralty-scale:source-reliability="f"	4
admiralty-scale:source-reliability="c"	4
veris:actor:motive="Financial"	4
ms-caro-malware:malware-platform="MacOS_X"	4
tor:tor-relay-type=exit-relay	4
circl:incident-classification=malware	4
misp-galaxy:tool="GAMEFISH"	3
misp-galaxy:ransomware="Dharma Ransomware"	3
misp-galaxy:mitre-intrusion-set="Dragonfly"	3
misp-galaxy:threat-actor="Cobalt"	3
misp-galaxy:tool="Mimikatz"	3
misp-galaxy:mitre-enterprise-attack-tool="Mimikatz - S0002"	3
misp-galaxy:banker="IcedID"	3
misp-galaxy:tool="PlugX"	3
misp-galaxy:tool="Chthonic"	3
misp-galaxy:tool="FINSPY"	3
misp-galaxy:ransomware="Matrix"	3
misp-galaxy:tool="Winnti"	3
misp-galaxy:exploit-kit="RIG"	3
misp-galaxy:tool="Snifula"	3
misp-galaxy:tool="Smoke Loader"	3
misp-galaxy:tool="Flokibot"	3
misp-galaxy:banker="Kronos"	3
misp-galaxy:mitre-mobile-attack-intrusion-set="APT28 - G0007"	3
circl:incident-classification="information-leak"	3

estimative-language:confidence-in-analytic-judgment="high"	3
circl:topic="industry"	3
enisa:nefarious-activity-abuse="exploits-exploit-kits"	3
malware_classification:malware-category="Spyware"	3
estimative-language:likelihood-probability="almost-certain"	3
veris:action:social:target="Finance"	3
circl:incident-classification="spam"	3
workflow:todo="add-tagging"	3
veris:action:social:variety="Scam"	3
veris:action:social:variety="Extortion"	3
misp-galaxy:tool="X-Tunnel"	2
misp-galaxy:mitre-malware="XAgentOSX"	2
misp-galaxy:threat-actor="Energetic Bear"	2
misp-galaxy:rat="FALLCHILL"	2
misp-galaxy:mitre-enterprise-attack-attack-pattern="Data from Local System - T1005"	2
misp-galaxy:ransomware="DBGer Ransomware"	2
misp-galaxy:banker="Trickbot"	2
misp-galaxy:rat="rokrat"	2
misp-galaxy:ransomware="Petya"	2
misp-galaxy:exploit-kit="Sundown"	2
misp-galaxy:mitre-enterprise-attack-attack-pattern="Exfiltration Over Command and Control Channel - T1041"	2
misp-galaxy:threat-actor="TERBIUM"	2
misp-galaxy:mitre-enterprise-attack-attack-pattern="Commonly Used Port - T1043"	2
misp-galaxy:threat-actor="Mirage"	2
misp-galaxy:ransomware="GandCrab"	2
misp-galaxy:threat-actor="NEODYMIUM"	2
misp-galaxy:mitre-intrusion-set="APT28"	2
misp-galaxy:tool="Satori"	2
misp-galaxy:tool="KHRAT"	2
misp-galaxy:threat-actor="APT 29"	2
misp-galaxy:tool="njRAT"	2
misp-galaxy:ransomware="Black Ruby"	2
misp-galaxy:tool="Zeus"	2
misp-galaxy:tool="gh0st"	2
misp-galaxy:banker="Geodo"	2
misp-galaxy:mitre-enterprise-attack-attack-pattern="Scripting - T1064"	2
misp-galaxy:android="Tizi"	2
misp-galaxy:threat-actor="PROMETHIUM"	2
misp-galaxy:mitre-enterprise-attack-attack-pattern="Exploit Public-Facing Application - T1190"	2
misp-galaxy:tool="Gafgyt"	2
misp-galaxy:threat-actor="Iron Group"	2
misp-galaxy:tool="TorrentLocker"	2
misp-galaxy:Ransomware="CryptoWall"	2
misp-galaxy:Ransomware="TeslaCrypt 0.x - 2.2.0"	2
ms-caro-malware-full:malware-type="Backdoor"	2
certsi:critical-sector="energy"	2
ms-caro-malware:malware-platform="Win64"	2
PasteBin: MALWAREMESSIAGH	2
estimative-language:confidence-in-analytic-judgment="low"	2
veris:action:social:vector="Documents"	2
workflow:todo="review-for-false-positive"	2
veris:asset:variety="S - SCADA"	2
dnc:malware-type="Ransomware"	2
europol-incident:availability="dos-ddos"	2
estimative-language:likelihood-probability="roughly-even-chance"	2
Banker	2

ecsirt:availability="ddos"	2
riskiq:threat-type="exploit-kit"	2
malware_classification:malware-category="Rootkit"	2
ms-caro-malware:malware-type="Trojan"	2
veris:actor:motive="Espionage"	2
expansion:whois-registrant-email	2
circl:incident-classification="cryptojacking"	2
veris:asset:variety="U - POS terminal"	2
ms-caro-malware:malware-type="DDoS"	2
Threat Type:APT	2
osint:source-type="source-code-repository"	2
ms-caro-malware:malware-platform="Python"	2
veris:action:malware:variety="Backdoor"	2
circl:topic="ict"	2
ms-caro-malware-full:malware-type="RemoteAccess"	2
ms-caro-malware-full:malware-family="ShellCode"	2
trickbot	2
workflow:todo="review"	2
riskiq:threat-type="scam"	2
riskiq:threat-name="scam-scareware"	2
osint:lifetime=perpetual	2
europol-event:brute-force-attempt=	2
osint:certainty=50	2
tor:tor-relay-type=	2

Appendix B

Private taxonomy mapping

Table B.1: Unified taxonomy mapping (detailed)

Unified taxonomy		Public taxonomies	Words
Tier1	Tier2		
abusive-content	spam	cccs:email-type="spam" circl:incident-classification="spam" ecsirt:abusive-content="spam" enisa:nefarious-activity-abuse="spam" europol-event:email-flooding europol-event:spam europol-incident:abusive-content="spam" gsma-fraud:technical="spamming" information-security-indicators:iex="spm.1" maec-malware-capabilities:maec-malware-capability="email-spam" rsit:abusive-content="spam" veris:action:malware:variety="spam" veris:action:social:variety="spam"	'spam', 'junk email', 'junk mail', 'junk e-mail', 'unsolicited email', 'unsolicited mail', 'unsolicited e-mail', 'bulk email', 'bulk mail', 'bulk e-mail', 'unwanted email', 'unwanted mail', 'unwanted e-mail'

malware	adware	cccs:malware-category="adware" malware_classification:malware-category="adware" ms-caro-malware:malware-type="adware" veris:action:malware:variety="adware"	‘adware’
	backdoor	maec-malware-behavior:maec-malware-behavior="install-backdoor" ms-caro-malware:malware-type="backdoor" ms-caro-malware-full:malware-type="backdoor" veris:action:malware:variety="backdoor"	‘backdoor’
	browser-modifier	cccs:malware-category="browser-hijacker" ms-caro-malware:malware-type="broswermodifier" ms-caro-malware-full:malware-type="broswermodifier"	'browser hijacker', 'browser modifier'
	cryptominer	circl:incident-classification="cryptojacking" maec-malware-behavior:maec-malware-behavior="mine-for-cryptocurrency" veris:action:malware:variety="click fraud"	'cryptominer', 'cryptojacking', 'cryptomining', 'cryptojacker', 'miner', 'mining'
	dialer	cert-xlm:malicious-code="dialer" ecsirt:malicious-code="dialer" ms-caro-malware:malware-type="dialer" ms-caro-malware-full:malware-type="dialer"	‘dialer’
	dos	maec-malware-behavior:maec-malware-behavior="denial-of-service" maec-malware-behavior:maec-malware-behavior="destroy-hardware" maec-malware-capabilities:maec-malware-capability="availability-violation" maec-malware-capabilities:maec-malware-capability="compromise-data-availability" maec-malware-capabilities:maec-malware-capability="compromise-system-availability" maec-malware-capabilities:maec-malware-capability="consume-system-resources" maec-malware-capabilities:maec-malware-capability="destruction" maec-malware-capabilities:maec-malware-capability="physical-entity-destruction" maec-malware-capabilities:maec-malware-capability="virtual-entity-destruction" ms-caro-malware:malware-type="ddoS" ms-caro-malware:malware-type="doS" ms-caro-malware-full:malware-type="ddos" ms-caro-malware-full:malware-type="dos" veris:action:malware:variety="doS"	'dos', 'ddos', 'destruction', 'destroy', 'destroying'

	exploit	cccs:malware-category="exploit-kit" enisa:nefarious-activity-abuse="exploits-exploit-kits" ms-caro-malware:malware-type="exploit" ms-caro-malware-full:malware-type="exploit" veris:action:malware:variety="exploit vuln" veris:action:malware:variety="sql injection"	'exploit'
	hack-tool	ms-caro-malware:malware-type="hacktool"	'hacktool', 'hack tool'
	misleading	circl:incident-classification="screenlocker" enisa:nefarious-activity-abuse="rogue-security-software-rogueware-scareware" ms-caro-malware:malware-type="joke" ms-caro-malware:malware-type="misleading" ms-caro-malware:malware-type="rogue" ms-caro-malware-full:malware-type="joke" ms-caro-malware-full:malware-type="misleading" ms-caro-malware-full:malware-type="rogue"	'joke', 'misleading', 'rogue', 'rogueware', 'scareware', 'screenlocker'
	monitoring-tool	maec-malware-behavior:maec-malware-behavior="capture maec-malware-capabilities:maec-malware-capability="discovery" maec-malware-capabilities:maec-malware-capability="network-environment-probing" ms-caro-malware:malware-type="monitoringtool" ms-caro-malware-full:malware-type="monitoringtool" veris:action:malware:variety="packet sniffer" veris:action:malware:variety="scan network"	'monitoring', 'monitor', 'scanning', 'scanner', 'sniffing', 'sniffer', 'probe', 'probing'
	password-stealer	enisa:nefarious-activity-abuse="credentials-stealing-trojans" maec-malware-behavior:maec-malware-behavior="crack-passwords" maec-malware-behavior:maec-malware-behavior="steal-password-hashes" maec-malware-behavior:maec-malware-behavior="steal-web-network-credential" maec-malware-capabilities:maec-malware-capability="authentication-credentials-theft" veris:action:malware:variety="password dumper"	'password stealer', 'credential stealer', 'password theft', 'credential theft', 'password stealing', 'credential stealing'

	ransomware	cccs:malware-category="ransomware" cert-xlm:malicious-code="ransomware" circl:incident-classification="locker" cryptocurrency-threat:crypto robbing ransomware ecsirt:malicious-code="ransomware" enisa:nefarious-activity-abuse="ransomware" maec-malware-behavior:maec-malware-behavior="encrypt-data" maec-malware-behavior:maec-malware-behavior="encrypt-files" malware_classification:malware-category="ransomware" ms-caro-malware:malware-type="ransom" ms-caro-malware-full:malware-type="ransom" veris:action:malware:variety="ransomware"	'ransom', 'ransomware'
	remote-access-tool	cccs:malware-category="webshell" ecsirt:malicious-code="botnet-drone" enisa:nefarious-activity-abuse="botnets-remote-activity" enisa:nefarious-activity-abuse="remote-access-tool" malware_classification:malware-category="botnet" ms-caro-malware:malware-type="remoteaccess" ms-caro-malware-full:malware-type="remoteaccess"	'remote access'
	settings-modifier	ecsirt:malicious-code="malware-configuration" ms-caro-malware:malware-type="settingsmodifier" ms-caro-malware-full:malware-type="settingsmodifier”	'settings modifier', 'setting modifier', 'configuration modifier', 'configurations modifier'
	spammer	maec-malware-capabilities:maec-malware-capability="email-spam" maec-malware-behavior:maec-malware-behavior="send-email-message" ms-caro-malware:malware-type="spammer" veris:action:malware:variety="spam"	'spammer', 'spam'
	spoofer	ms-caro-malware:malware-type="spoofer" ms-caro-malware-full:malware-type="spoofer"	'spoofer', 'spoofing'

	spyware	cccs:malware-category="keylogger" cccs:malware-category="spyware" cert-xlm:malicious-code="spyware-rat" ecsirt:malicious-code="spyware" malware_classification:malware-category="spyware" ms-caro-malware:malware-type="spyware" ms-caro-malware-full:malware-type="spyware veris:action:malware:variety="spyware/keylogger"	'spyware', 'keylogger'
	trojan	cccs:malware-category="trojan" cert-xlm:malicious-code="trojan-malware" ecsirt:malicious-code="trojan" malware_classification:malware-category="downloader" malware_classification:malware-category="trojan" ms-caro-malware:malware-type="trojan" ms-caro-malware:malware-type="trojanclicker" ms-caro-malware:malware-type="trojandownloader" ms-caro-malware:malware-type="trojandropper" ms-caro-malware:malware-type="trojannotifier" ms-caro-malware:malware-type="trojanproxy" ms-caro-malware:malware-type="trojanspy" ms-caro-malware-full:malware-type="trojan" ms-caro-malware-full:malware-type="trojanclicker" ms-caro-malware-full:malware-type="trojandownloader"	'trojan', 'trojanclicker', 'trojandownloader', 'trojandropper', ‘clicker', 'downloader', 'dropper'
	virtool	cccs:malware-category="rootkit" cert-xlm:malicious-code="rootkit" ecsirt:malicious-code="rootkit" enisa:nefarious-activity-abuse="rootkits" malware_classification:malware-category="rootkit" ms-caro-malware:malware-type="virtool" ms-caro-malware-full:malware-type="virtool" veris:action:malware:variety="rootkit"	'rootkit', 'rootkits', 'virtool'

	virus	cccs:malware-category="virus" cert-xlm:malicious-code="virus" ecsirt:malicious-code="virus" enisa:nefarious-activity-abuse="viruses" malware_classification:malware-category="virus" ms-caro-malware:malware-type="virus" ms-caro-malware-full:malware-type="virus"	'virus', 'viruses'
	wiper	circl:incident-classification="wiper" maec-malware-behavior:maec-malware-behavior="erase-data" maec-malware-capabilities:maec-malware-capability="system-operational-integrity-violation" veris:action:malware:variety="destroy data" veris:action:malware:variety="destroy data"	'wiper', 'erasure', 'erase', 'wipe', 'wiping', 'erasing'
	worm	cccs:malware-category="worm" cert-xlm:malicious-code="worm" ecsirt:malicious-code="worm" enisa:nefarious-activity-abuse="worms-trojans" europol-event:worm-spreading malware_classification:malware-category="worm" ms-caro-malware:malware-type="worm" ms-caro-malware-full:malware-type="worm" veris:action:malware:variety="worm"	'worm', 'worms'
information-gathering	scanning	cccs:scan-type=* cert-xlm:information-gathering="scanner" circl:incident-classification="scan" ecsirt:information-gathering="scanner" europol-event:network-scanning europol-incident:information-gathering="scanning" incident-disposition:not-an-incident="scan-probe" pentest:approach="vulnerability_scanning" pentest:network="network_discovery" pentest:scan=* veris:action:malware:variety="scan network"	'scanning', 'scan', 'scanner'

	sniffing	cert-xlm:information-gathering="sniffing" ecsirt:information-gathering="sniffing" pentest:network="sniffing" veris:action:physical:variety="snooping" veris:action:physical:variety="surveillance" veris:action:physical:variety="wiretapping"	'wiretapping', 'monitoring'
	social-engineering	cert-xlm:information-gathering="social-engineering" ecsirt:information-gathering="social-engineering" enisa:nefarious-activity-abuse="social-engineering" gsma-fraud:business="social-engineering" information-security-indicators:idb="rgh.2" information-security-indicators:vbh="huw.2" open_threat:threat-name="per-004" smart-airports-threats:malicious-actions="social-attacks-baiting" smart-airports-threats:malicious-actions="social-attacks-impersonation" smart-airports-threats:malicious-actions="social-attacks-pretexting" smart-airports-threats:malicious-actions="social-attacks-reverse-social-engineering" veris:action:social:variety="baiting" veris:action:social:variety="bribery" veris:action:social:variety="elicitation" veris:action:social:variety="influence" veris:action:social:variety="pretexting" veris:action:social:variety="propaganda"	'social', 'engineering', 'personnel behaviour', 'impersonation', 'impersonations', 'impersonating', 'trick', 'tricks', 'tricking', 'deception', 'deceptions', 'elicitation'
intrusion-or-attempts	ids-alert	cert-xlm:intrusion-attempts="exploit-known-vuln" ecsirt:intrusion-attempts="ids-alert" europol-event:brute-force-attempt europol-event:dictionary-attack-attempt europol-event:exploit-attempt europol-event:file-inclusion-attempt europol-event:password-cracking-attempt europol-event:sql-injection-attempt europol-event:xss-attempt europol-incident:intrusion-attempt="exploitation-vulnerability" information-security-indicators:iex="int.1" rsit:intrusion-attempts="ids-alert"	'attempt to compromise', 'attempted compromise', 'attempt to exploit', 'attempted exploit', 'attempt exploitation'

	brute-force	cert-xlm:intrusion-attempts="login-attempts" ecsirt:intrusion-attempts="brute-force" europol-event:brute-force-attempt europol-event:dictionary-attack-attempt europol-event:password-cracking-attempt europol-incident:intrusion-attempt="login-attempt" pentest:web="bruteforce" rsit:intrusion-attempts="brute-force" veris:action:hacking:variety="Brute force"	'brute', 'login attempt', 'login attempts'
	unknown-exploit	cccs:exploitation-technique="other" cert-xlm:intrusion-attempts="new-attack-signature" ecsirt:intrusion-attempts="exploit"	'unknown exploit', 'new attack', 'new signature'
	account-compromise	cert-xlm:intrusion="privileged-account-compromise" cert-xlm:intrusion="unprivileged-account-compromise" ecsirt:intrusions="privileged-account-compromise" ecsirt:intrusions="unprivileged-account-compromise" rsit:intrusions="privileged-account-compromise" rsit:intrusions="unprivileged-account-compromise"	'account compromise', 'credentials compromise', 'successful login', 'login with success', 'authenticated with success', 'successful authentication'

	system-or-application-compromise	cccs:exploitation-technique=* cert-xlm:intrusion="application-compromise" cert-xlm:intrusion="domain-compromise" circl:incident-classification="sql-injection" circl:incident-classification="system-compromise" circl:incident-classification="XSS" ecsirt:intrusions="application-compromise" ecsirt:intrusions="backdoor" ecsirt:intrusions="compromised" ecsirt:intrusions="defacement" enisa:nefarious-activity-abuse="web-application-attacks-injection-attacks-code-injection-SQL-XSS" europol-event:control-system-bypass europol-event:exploit europol-event:file-inclusion europol-event:sql-injection europol-event:theft-access-credentials europol-event:unauthorized-access-system europol-event:xss infoleak:analyst-detection="sql-injection" infoleak:automatic-detection="sql-injection" information-security-indicators:iex="int.2 information-security-indicators:iex="int.3 pentest:exploit=* pentest:web=* rsit:intrusions="application-compromise" veris:action:hacking:variety="Brute force" veris:action:hacking:variety="Buffer overflow" veris:action:hacking:variety="Cache poisoning" veris:action:hacking:variety="CSRF" veris:action:hacking:variety="Format string attack" veris:action:hacking:variety="Fuzz testing" veris:action:hacking:variety="HTTP request smuggling" veris:action:hacking:variety="HTTP request splitting" veris:action:hacking:variety="HTTP response smuggling" veris:action:hacking:variety="HTTP Response Splitting"	'domain compromise', 'application compromise', 'system compromise', 'domain intrusion', 'application intrusion', 'system intrusion'

		veris:action:hacking:variety="Integer overflows" veris:action:hacking:variety="LDAP injection" veris:action:hacking:variety="Mail command injection" veris:action:hacking:variety="MitM" veris:action:hacking:variety="Null byte injection" veris:action:hacking:variety="OS commanding" veris:action:hacking:variety="Pass-the-hash" veris:action:hacking:variety="Path traversal" veris:action:hacking:variety="RFI" veris:action:hacking:variety="Session fixation" veris:action:hacking:variety="Session prediction" veris:action:hacking:variety="Session replay" veris:action:hacking:variety="Soap array abuse" veris:action:hacking:variety="Special element injection" veris:action:hacking:variety="SQLi" veris:action:hacking:variety="SSI injection" veris:action:hacking:variety="URL redirector abuse" veris:action:hacking:variety="XML attribute blowup" veris:action:hacking:variety="XML entity expansion" veris:action:hacking:variety="XML external entities" veris:action:hacking:variety="XML injection" veris:action:hacking:variety="XPath injection" veris:action:hacking:variety="XQuery injection" veris:action:hacking:variety="XSS"	
	botnet-member	cert-xlm:intrusion="botnet-member" ecsirt:intrusions="bot"	'bot', 'botnet member'

availability	dos-or-ddos	cccs:event="dos" circl:incident-classification="denial-of-service" csirt_case_classification:incident-category="DOS" ddos:type="amplification-attack" ddos:type="flooding-attack" ddos:type="post-attack" ddos:type="reflected-spoofed-attack" ddos:type="slow-read-attack" ecsirt:availability="ddos" ecsirt:availability="dos enisa:nefarious-activity-abuse="denial-of-service" enisa:nefarious-activity-abuse="distributed-denial-of-network-service-amplification-reflection-attack" enisa:nefarious-activity-abuse="distributed-denial-of-network-service-application-layer-attack" enisa:nefarious-activity-abuse="distributed-denial-of-network-service-network-layer-attack" europol-incident:availability="dos-ddos" information-security-indicators:IEX="DOS.1" ms-caro-malware:malware-type="DDoS" ms-caro-malware:malware-type="DoS" ms-caro-malware-full:malware-type="DDoS" ms-caro-malware-full:malware-type="DoS" rsit:availability="ddos" rsit:availability="dos" veris:action:hacking:variety="DoS" veris:action:malware:variety="DoS"	'dos', 'ddos', 'denial of service', 'disruption', 'degradation', 'exhaustion'
information-content-security	unauthorised-information-access	cert-xlm:information-content-security="unauthorised-information-access" common-taxonomy:information-security="unauthorised-access" ecsirt:information-content-security="unauthorised-information-access" enisa:physical-attack="information-leak-or-unauthorised-sharing" enisa:physical-attack="unauthorised-physical-access-or-unauthorised-entry-to-premises" europol-event:unauthorized-access-information europol-incident:information-security="unauthorized-access" monarc-threat:unauthorised-actions="illegal-processing-of-data" rsit:information-content-security="unauthorised-information-access"	'unauthorised access', 'unauthorised information access', 'unauthorised data access'

	unauthorised-information-modification	cert-xlm:information-content-security="unauthorised-information-modification" common-taxonomy:information-security="unauthorised-modification-or-deletion" ecsirt:information-content-security="unauthorised-information-modification" enisa:nefarious-activity-abuse="unauthorized-changes-of-records" europol-event:deletion-information europol-event:modification-information europol-incident:information-security="unauthorized-modification" monarc-threat:unauthorised-actions="corruption-of-data" rsit:information-content-security="unauthorised-information-modification"	'unauthorised modification', 'unauthorised information modification', 'unauthorised data modification'
fraud	masquerade	cert-xlm:fraud="masquerade" ecsirt:fraud="masquerade" enisa:nefarious-activity-abuse="identity-theft-identity-fraud-account information-security-indicators:idb="uid.1" monarc-threat:compromise-of-functions="forging-of-rights" rsit:fraud="masquerade"	'masquerade', 'forged identity'

	phishing	cccs:email-type="phishing" cccs:event="phishing" circl:incident-classification="phishing" common-taxonomy:information-gathering="phishing" ecsirt:fraud="phishing" enisa:nefarious-activity-abuse="phishing-attacks" enisa:nefarious-activity-abuse="spear-phishing-attacks" enisa:nefarious-activity-abuse="spear-phishing-attacks-targeted" europol-event:aggregation-information-phishing-schemes europol-event:dissemination-phishing-emails europol-event:hosting-phishing-sites europol-incident:information-gathering="phishing" gsma-fraud:technical="phishing-pharming" information-security-indicators:iex="fgy.2" information-security-indicators:iex="phi.1" information-security-indicators:iex="phi.2" information-security-indicators:vbh="huw.1" maec-delivery-vectors:maec-delivery-vector="pharming" maec-delivery-vectors:maec-delivery-vector="phishing" pentest:social_engineering="phishing" rsit:fraud="phishing" smart-airports-threats:malicious-actions="social-attacks-phishing-spearphishing" veris:action:social:variety="phishing"	'phishing', 'pharming', 'spearphishing', 'whaling'
vulnerable	vulnerable-service	cccs:misusage-type="vulnerable-software" CERT-XLM:vulnerable="vulnerable-service" ecsirt:vulnerable="vulnerable-service" rsit:vulnerable="vulnerable-service"	'vulnerable', 'vulnerability'

Appendix C

Attribute group distribution

Table C.1: Attribute group distribution for abusive-content

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[	]100,+∞[	]500,+∞[	]1000,+∞[
URL	29,7%	25,0%	22,2%	16,8%	12,8%	4,3%	4,6%
Network address	27,7%	25,8%	29,4%	24,8%	24,0%	23,4%	14,6%
Network name	27,1%	22,9%	20,4%	15,5%	11,9%	4,3%	4,6%
File hash	7,5%	13,7%	14,6%	22,9%	27,7%	36,9%	41,5%
Other Info	3,1%	5,5%	6,0%	8,4%	10,1%	12,9%	13,9%
File sample	2,4%	5,9%	6,4%	10,8%	13,4%	18,2%	20,7%
File name	1,5%	0,7%	0,6%	0,4%	0,1%	0,0%	0,0%
Email text	0,5%	0,2%	0,2%	0,1%	0,0%	0,0%	0,0%
Agent	0,2%	0,1%	0,1%	0,1%	0,0%	0,0%	0,0%
Date	0,2%	0,1%	0,1%	0,0%	0,0%	0,0%	0,0%
Email address	0,1%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Email other info	0,1%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Regkey	0,1%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Bank account	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Bank id	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Bank other info	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Email name	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
File other info	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Location	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Mac address	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Network hash	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Network id	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Network request	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Organization	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Pattern	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Personal id	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Personal location	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Personal name	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Personal other info	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Phone number	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Process name	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Process other info	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Rule	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Threat actor	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Travel	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
URI	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Vulnerability	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
X509 fingerprint	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%

Table C.2: Attribute group distribution for malicious-code

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[	]100,+∞[	]500,+∞[	]1000,+∞[
File hash	24,2%	28,7%	32,8%	32,4%	35,7%	38,4%	31,3%
URL	17,3%	15,4%	13,2%	10,2%	7,3%	1,7%	1,9%
Network address	17,2%	15,5%	15,1%	12,5%	10,6%	7,7%	5,3%
Network name	16,2%	14,6%	13,4%	21,2%	23,1%	31,7%	42,1%
Other Info	15,4%	15,9%	16,0%	14,5%	14,2%	12,3%	10,5%
File name	3,0%	3,2%	2,8%	2,1%	1,7%	0,3%	0,0%
Date	1,5%	2,4%	2,4%	1,7%	1,8%	0,6%	0,0%
File sample	1,1%	2,0%	2,0%	3,9%	5,0%	6,9%	8,9%
Email address	1,0%	0,5%	0,4%	0,3%	0,1%	0,0%	0,0%
Bank account	0,7%	0,5%	0,4%	0,3%	0,1%	0,0%	0,0%
Regkey	0,5%	0,3%	0,2%	0,2%	0,0%	0,0%	0,0%
Rule	0,5%	0,3%	0,2%	0,2%	0,0%	0,0%	0,0%
File other info	0,3%	0,1%	0,2%	0,2%	0,1%	0,2%	0,0%
Email text	0,2%	0,1%	0,1%	0,1%	0,0%	0,0%	0,0%
Process other info	0,2%	0,1%	0,2%	0,1%	0,1%	0,1%	0,0%
Vulnerability	0,2%	0,1%	0,1%	0,1%	0,0%	0,0%	0,0%
Agent	0,1%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Organization	0,1%	0,1%	0,0%	0,0%	0,0%	0,0%	0,0%
Pattern	0,1%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Personal name	0,1%	0,1%	0,0%	0,0%	0,0%	0,0%	0,0%
Threat actor	0,1%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
X509 fingerprint	0,1%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Bank id	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Bank other info	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Email name	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Email other info	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Location	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Mac address	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Network hash	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Network id	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Network request	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Personal id	0,0%	0,1%	0,0%	0,0%	0,0%	0,0%	0,0%
Personal location	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Personal other info	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Phone number	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Process name	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Travel	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
URI	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%

Table C.3: Attribute group distribution for information-gathering

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[	]100,+∞[	]500,+∞[	]1000,+∞[
Network address	35,2%	24,7%	24,7%	12,7%	8,2%	0,7%	0,7%
File hash	22,2%	22,8%	22,8%	11,4%	9,3%	0,0%	0,0%
Other Info	12,3%	10,2%	10,2%	5,1%	3,7%	0,0%	0,0%
URL	12,3%	12,4%	12,4%	6,2%	5,0%	0,0%	0,0%
Network name	12,1%	22,9%	22,9%	60,9%	70,9%	99,2%	99,2%
File name	1,8%	3,1%	3,1%	1,5%	1,5%	0,0%	0,0%
Vulnerability	0,9%	0,3%	0,3%	0,1%	0,0%	0,0%	0,0%
Email text	0,8%	0,3%	0,3%	0,1%	0,0%	0,0%	0,0%
Bank account	0,4%	1,1%	1,1%	0,6%	0,6%	0,0%	0,0%
Date	0,3%	1,5%	1,5%	0,7%	0,8%	0,0%	0,0%
File other info	0,3%	0,1%	0,1%	0,1%	0,0%	0,0%	0,0%
Personal location	0,3%	0,1%	0,1%	0,1%	0,0%	0,0%	0,0%
Phone number	0,3%	0,1%	0,1%	0,0%	0,0%	0,0%	0,0%
X509 fingerprint	0,3%	0,1%	0,1%	0,0%	0,0%	0,0%	0,0%
Email address	0,2%	0,2%	0,2%	0,1%	0,1%	0,0%	0,0%
File sample	0,2%	0,1%	0,1%	0,0%	0,0%	0,0%	0,0%
Personal name	0,1%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Process other info	0,1%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Rule	0,1%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Agent	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Bank id	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Bank other info	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Email name	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Email other info	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Location	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Mac address	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Network hash	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Network id	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Network request	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Organization	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Pattern	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Personal id	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Personal other info	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Process name	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Regkey	0,0%	0,2%	0,2%	0,1%	0,1%	0,0%	0,0%
Threat actor	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
Travel	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%
URI	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%	0,0%

Table C.4: Attribute group distribution for intrusion-or-intrusion-attempts

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[	]100,+∞[	]500,+∞[	]1000,+∞[
Other Info	31%	23%	10%	10%	7%	0%	NDA
File hash	30%	31%	13%	13%	11%	1%	NDA
Network name	22%	7%	6%	6%	4%	6%	NDA
Date	7%	7%	3%	3%	3%	0%	NDA
File name	4%	3%	1%	1%	1%	0%	NDA
Network address	3%	27%	54%	54%	62%	74%	NDA
URL	3%	2%	11%	11%	13%	18%	NDA
Email address	1%	0%	0%	0%	0%	0%	NDA
Rule	0%	0%	0%	0%	0%	0%	NDA
Email text	0%	0%	0%	0%	0%	0%	NDA
File sample	0%	0%	0%	0%	0%	0%	NDA
Process other info	0%	0%	0%	0%	0%	0%	NDA
Regkey	0%	0%	0%	0%	0%	0%	NDA
Agent	0%	0%	0%	0%	0%	0%	NDA
Bank account	0%	0%	0%	0%	0%	0%	NDA
Bank id	0%	0%	0%	0%	0%	0%	NDA
Bank other info	0%	0%	0%	0%	0%	0%	NDA
Email name	0%	0%	0%	0%	0%	0%	NDA
Email other info	0%	0%	0%	0%	0%	0%	NDA
File other info	0%	0%	0%	0%	0%	0%	NDA
Location	0%	0%	0%	0%	0%	0%	NDA
Mac address	0%	0%	0%	0%	0%	0%	NDA
Network hash	0%	0%	0%	0%	0%	0%	NDA
Network id	0%	0%	0%	0%	0%	0%	NDA
Network request	0%	0%	0%	0%	0%	0%	NDA
Organization	0%	0%	0%	0%	0%	0%	NDA
Pattern	0%	0%	0%	0%	0%	0%	NDA
Personal id	0%	0%	0%	0%	0%	0%	NDA
Personal location	0%	0%	0%	0%	0%	0%	NDA
Personal name	0%	0%	0%	0%	0%	0%	NDA
Personal other info	0%	0%	0%	0%	0%	0%	NDA
Phone number	0%	0%	0%	0%	0%	0%	NDA
Process name	0%	0%	0%	0%	0%	0%	NDA
Threat actor	0%	0%	0%	0%	0%	0%	NDA
Travel	0%	0%	0%	0%	0%	0%	NDA
URI	0%	0%	0%	0%	0%	0%	NDA
Vulnerability	0%	0%	0%	0%	0%	0%	NDA
X509 fingerprint	0%	0%	0%	0%	0%	0%	NDA

Table C.5: Attribute group distribution for availability

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[	]100,+∞[	]500,+∞[	]1000,+∞[
Network name	33,3%	33,3%	33,3%	33,3%	NDA	NDA	NDA
Network address	24,8%	24,8%	24,8%	24,8%	NDA	NDA	NDA
Other Info	22,9%	22,9%	22,9%	22,9%	NDA	NDA	NDA
File hash	14,3%	14,3%	14,3%	14,3%	NDA	NDA	NDA
Rule	1,9%	1,9%	1,9%	1,9%	NDA	NDA	NDA
Date	1,0%	1,0%	1,0%	1,0%	NDA	NDA	NDA
File name	1,0%	1,0%	1,0%	1,0%	NDA	NDA	NDA
URL	1,0%	1,0%	1,0%	1,0%	NDA	NDA	NDA
Agent	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Bank account	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Bank id	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Bank other info	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Email address	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Email name	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Email other info	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Email text	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
File other info	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
File sample	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Location	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Mac address	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Network hash	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Network id	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Network request	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Organization	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Pattern	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Personal id	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Personal location	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Personal name	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Personal other info	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Phone number	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Process name	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Process other info	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Regkey	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Threat actor	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Travel	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
URI	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Vulnerability	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
X509 fingerprint	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA

Table C.6: Attribute group distribution for information-content-security

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[	]100,+∞[	]500,+∞[	]1000,+∞[
Other Info	52,0%	52,0%	52,0%	52,0%	NDA	NDA	NDA
File name	29,3%	29,3%	29,3%	29,3%	NDA	NDA	NDA
File hash	10,7%	10,7%	10,7%	10,7%	NDA	NDA	NDA
Date	2,7%	2,7%	2,7%	2,7%	NDA	NDA	NDA
File sample	1,3%	1,3%	1,3%	1,3%	NDA	NDA	NDA
Network address	1,3%	1,3%	1,3%	1,3%	NDA	NDA	NDA
Regkey	1,3%	1,3%	1,3%	1,3%	NDA	NDA	NDA
URL	1,3%	1,3%	1,3%	1,3%	NDA	NDA	NDA
Agent	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Bank account	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Bank id	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Bank other info	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Email address	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Email name	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Email other info	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Email text	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
File other info	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Location	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Mac address	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Network hash	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Network id	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Network name	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Network request	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Organization	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Pattern	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Personal id	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Personal location	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Personal name	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Personal other info	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Phone number	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Process name	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Process other info	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Rule	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Threat actor	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Travel	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
URI	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
Vulnerability	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA
X509 fingerprint	0,0%	0,0%	0,0%	0,0%	NDA	NDA	NDA

Table C.7: Attribute group distribution for fraud

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[	]100,+∞[	]500,+∞[	]1000,+∞[
Network name	50%	49%	58%	81%	83%	91%	99%
File hash	14%	23%	13%	6%	5%	0%	0%
URL	11%	4%	5%	2%	2%	2%	0%
Other Info	11%	9%	11%	5%	5%	4%	0%
Email address	5%	1%	3%	1%	1%	1%	0%
Network address	4%	5%	3%	2%	1%	1%	1%
Rule	2%	1%	0%	0%	0%	0%	0%
File name	1%	3%	2%	1%	1%	0%	0%
Vulnerability	1%	0%	0%	0%	0%	0%	0%
Date	0%	3%	4%	2%	2%	2%	0%
Email text	0%	0%	0%	0%	0%	0%	0%
Personal name	0%	0%	0%	0%	0%	0%	0%
Regkey	0%	0%	0%	0%	0%	0%	0%
X509 fingerprint	0%	0%	0%	0%	0%	0%	0%
Agent	0%	0%	0%	0%	0%	0%	0%
Bank account	0%	0%	0%	0%	0%	0%	0%
Bank id	0%	0%	0%	0%	0%	0%	0%
Bank other info	0%	0%	0%	0%	0%	0%	0%
Email name	0%	0%	0%	0%	0%	0%	0%
Email other info	0%	0%	0%	0%	0%	0%	0%
File other info	0%	0%	0%	0%	0%	0%	0%
File sample	0%	0%	0%	0%	0%	0%	0%
Location	0%	0%	0%	0%	0%	0%	0%
Mac address	0%	0%	0%	0%	0%	0%	0%
Network hash	0%	0%	0%	0%	0%	0%	0%
Network id	0%	0%	0%	0%	0%	0%	0%
Network request	0%	0%	0%	0%	0%	0%	0%
Organization	0%	0%	0%	0%	0%	0%	0%
Pattern	0%	0%	0%	0%	0%	0%	0%
Personal id	0%	1%	1%	0%	0%	0%	0%
Personal location	0%	0%	0%	0%	0%	0%	0%
Personal other info	0%	0%	0%	0%	0%	0%	0%
Phone number	0%	0%	0%	0%	0%	0%	0%
Process name	0%	0%	0%	0%	0%	0%	0%
Process other info	0%	0%	0%	0%	0%	0%	0%
Threat actor	0%	0%	0%	0%	0%	0%	0%
Travel	0%	0%	0%	0%	0%	0%	0%
URI	0%	0%	0%	0%	0%	0%	0%

Table C.8: Attribute group distribution for vulnerable

Group	]0,100]	]0,500]	]0,1000]	]0,+∞[	]100,+∞[	]500,+∞[	]1000,+∞[
File hash	53%	53%	53%	53%	NDA	NDA	NDA
Other Info	18%	18%	18%	18%	NDA	NDA	NDA
File name	13%	13%	13%	13%	NDA	NDA	NDA
Network name	11%	11%	11%	11%	NDA	NDA	NDA
Rule	3%	3%	3%	3%	NDA	NDA	NDA
Network address	2%	2%	2%	2%	NDA	NDA	NDA
Process other info	1%	1%	1%	1%	NDA	NDA	NDA
Agent	0%	0%	0%	0%	NDA	NDA	NDA
Bank account	0%	0%	0%	0%	NDA	NDA	NDA
Bank id	0%	0%	0%	0%	NDA	NDA	NDA
Bank other info	0%	0%	0%	0%	NDA	NDA	NDA
Date	0%	0%	0%	0%	NDA	NDA	NDA
Email address	0%	0%	0%	0%	NDA	NDA	NDA
Email name	0%	0%	0%	0%	NDA	NDA	NDA
Email other info	0%	0%	0%	0%	NDA	NDA	NDA
Email text	0%	0%	0%	0%	NDA	NDA	NDA
File other info	0%	0%	0%	0%	NDA	NDA	NDA
File sample	0%	0%	0%	0%	NDA	NDA	NDA
Location	0%	0%	0%	0%	NDA	NDA	NDA
Mac address	0%	0%	0%	0%	NDA	NDA	NDA
Network hash	0%	0%	0%	0%	NDA	NDA	NDA
Network id	0%	0%	0%	0%	NDA	NDA	NDA
Network request	0%	0%	0%	0%	NDA	NDA	NDA
Organization	0%	0%	0%	0%	NDA	NDA	NDA
Pattern	0%	0%	0%	0%	NDA	NDA	NDA
Personal id	0%	0%	0%	0%	NDA	NDA	NDA
Personal location	0%	0%	0%	0%	NDA	NDA	NDA
Personal name	0%	0%	0%	0%	NDA	NDA	NDA
Personal other info	0%	0%	0%	0%	NDA	NDA	NDA
Phone number	0%	0%	0%	0%	NDA	NDA	NDA
Process name	0%	0%	0%	0%	NDA	NDA	NDA
Regkey	0%	0%	0%	0%	NDA	NDA	NDA
Threat actor	0%	0%	0%	0%	NDA	NDA	NDA
Travel	0%	0%	0%	0%	NDA	NDA	NDA
URI	0%	0%	0%	0%	NDA	NDA	NDA
URL	0%	0%	0%	0%	NDA	NDA	NDA
Vulnerability	0%	0%	0%	0%	NDA	NDA	NDA
X509 fingerprint	0%	0%	0%	0%	NDA	NDA	NDA

Appendix D

Events that form the Cluster in Section 6.4

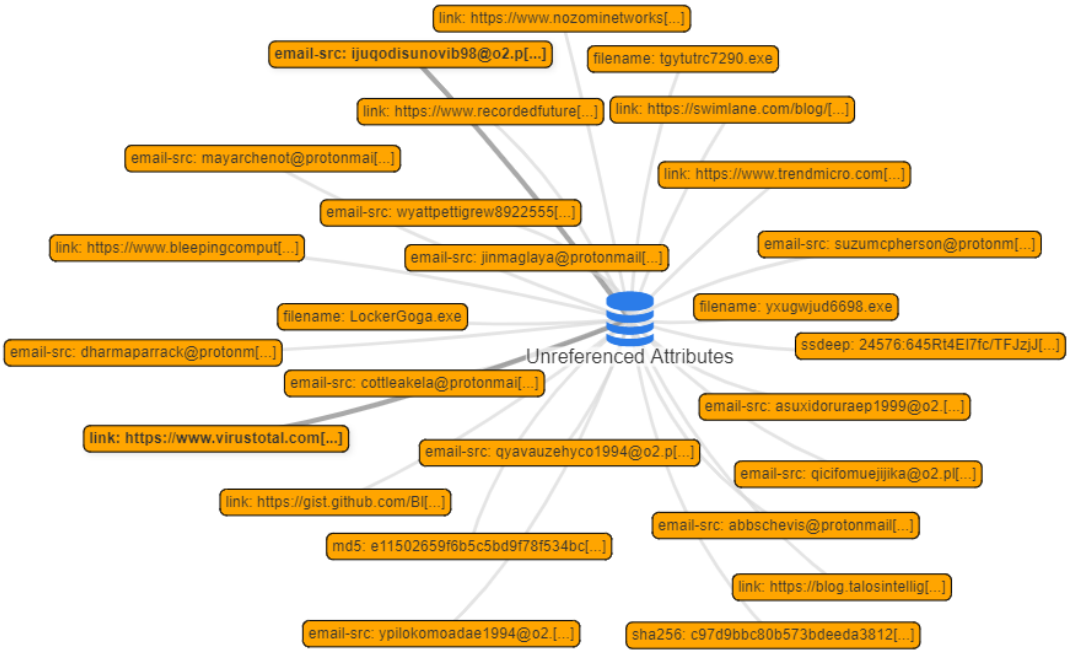

Figure D.1: Event 1518 before being processed by AECCP

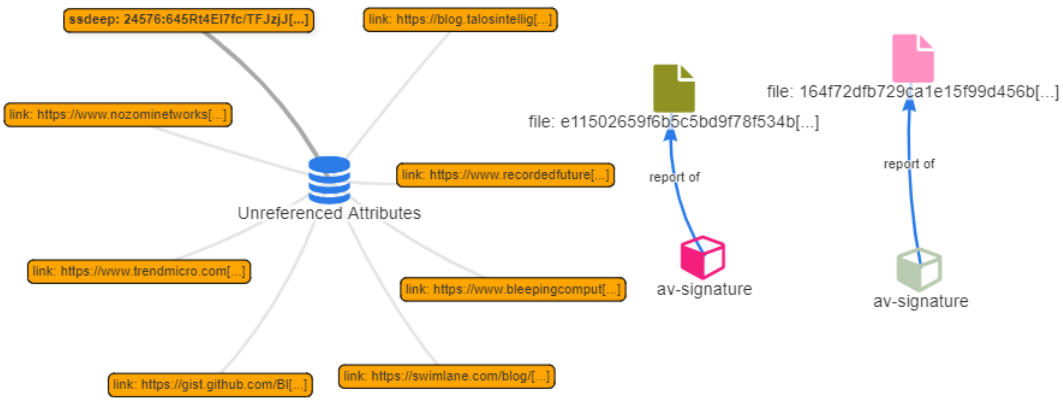

Figure D.2: Event 1518 after being processed by the Enricher

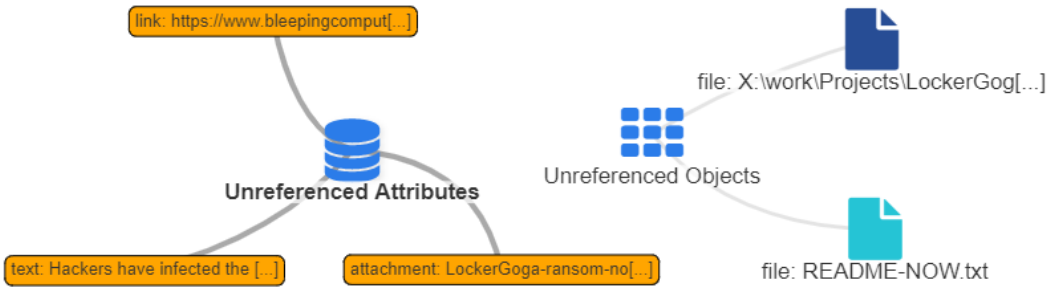

Figure D.3: Event 1520 after being processed by the Enricher

Bibliography

[1] Symantec World Headquarters, “Advanced Persistent Threats: A Symantec Perspective,” pp. http://index-of.es/Varios/b-advanced_persistent_threats_WP_21215957.en-us.pdf, 2011.

[2] SWIFT, “The Evolving Cyber Threat,” 2017. [Online]. Available: https://www.swift.com/pt/node/147646.

[3] “Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC,” [Online]. Available: https://eur-lex.europa.eu/eli/reg/2016/679/oj.

[4] R. M. Lee, “Intelligence Defined and its Impact on Cyber Threat Intelligence,” [Online]. Available: https://www.robertmlee.org/intelligence-defined-and-its-impact-on-cyber-threat-intelligence/. [Accessed November 2018].

[5] FireEye Inc., “Taking a Lean-Forward Approach to Combat Today's Cyber Attacks,” 2014. [Online]. Available: https://informationsecurity.report/Resources/Whitepapers/baab01e7-e39a-4c17-9562-bd3753cca903_Taking%20a%20Lean-Forward%20Approach%20to%20Combat%20Today's%20Cyber%20Attacks.pdf.

[6] W. Tounsi e H. Rais, “A survey on technical threat intelligence in the age of sophisticated cyber attacks,” 2018. [Online]. Available: https://www.researchgate.net/publication/320027747_A_survey_on_technical_threat_intelligence_in_the_age_of_sophisticated_cyber_attacks.

[7] C. Martins e I. Medeiros, “Generating Threat Intelligence by Classification and Association of Security Events,” 2019. [Online]. Available: http://www.di.fc.ul.pt/~imedeiros/papers/DSN2019_DCDS_IoC_Class.pdf.

[8] P. Chen, L. Desmet e C. Huygens, “A Study on Advanced Persistent Threats,” 2016. [Online]. Available: https://hal.inria.fr/hal-01404186/document.

[9] T. Yadav e R. A. Mallari, “Technical Aspects of Cyber Kill Chain,” 2015. [Online]. Available: https://arxiv.org/pdf/1606.03184.pdf.

[10] M. Glassman e M. J. Kang, “Intelligence in the internet age: The emergence and evolution of Open,” 2012. [Online]. Available: https://www.researchgate.net/publication/220495751_Intelligence_in_the_internet_age_The_emergence_and_evolution_of_Open_Source_Intelligence_OSINT.

[11] Q. Eijkman e D. Weggemans, “Open Source Intelligence and Privacy Dilemmas: Is it Time to Reassess State Accountability?,” 2013. [Online]. Available: https://cyberwar.nl/d/03_Eijkman_Weggemans_v2%5B1%5D_1367418023.pdf.

[12] J. Pastor-Galindo, P. Nespoli, F. G. Mármol e G. M. P. , “The not yet exploited goldmine of OSINT: Opportunities, open challenges and future trends,” 2020. [Online]. Available: https://ieeexplore.ieee.org/document/8954668.

[13] G. Hribar, I. Podbregar e T. Ivanuša, “OSINT: A “Grey Zone”,” 2014. [Online]. Available: https://www.researchgate.net/publication/262581640_OSINT_A_Grey_Zone.

[14] Webroot, “Threat Intelligence: What is it, and How Can it Protect You from Today s Advanced Cyber-Attacks,” [Online]. Available: https://www.gartner.com/imagesrv/media-products/pdf/webroot/issue1 webroot.pdf.

[15] M. Bromiley, “Threat Intelligence: What It Is, and How to Use It Effectively,” 2016. [Online]. Available: https://www.sans.org/reading-room/whitepapers/threathunting/paper/37282.

[16] R. M. Lee, “2020 SANS Cyber Threat Intelligence (CTI) Survey,” 2020. [Online]. Available: https://www.sans.org/reading-room/whitepapers/threats/paper/39395.

[17] B. P. Kime, “Threat Intelligence: Planning and Direction,” 2019. [Online]. Available: https://www.sans.org/reading-room/whitepapers/threatintelligence/threat-intelligence-planning-direction-36857.

[18] T. MATTERN, J. FELKER, R. BORUM e G. BAMFORD, “Operational Levels of Cyber Intelligence,” p. https://www.researchgate.net/publication/264562681_Operational_Levels_of_Cyber_Intelligence, 2014.

[19] Bank of England, “Understanding Cyber Threat Intelligence Operations,” 2016. [Online]. Available: https://www.bankofengland.co.uk/-/media/boe/files/financial-stability/financial-sector-continuity/understanding-cyber-threat-intelligence-operations.pdf.

[20] ENISA, “Standards and tools for exchange and processing of actionable information,” 2015. [Online]. Available: https://www.enisa.europa.eu/publications/standards-and-tools-for-exchange-and-processing-of-actionable-information.

[21] A. Ramsdale, S. Shiaeles e N. Kolokotronis, “A Comparative Analysis of Cyber-Threat Intelligence Sources, Formats and Languages,” 2020. [Online]. Available: https://www.mdpi.com/2079-9292/9/5/824.

[22] S. Barnum, “Standardizing Cyber Threat Intelligence Information with the Structured Threat Information eXpression (STIX™),” 2014. [Online]. Available: https://www.mitre.org/publications/technical-papers/standardizing-cyber-threat-intelligence-information-with-the.

[23] R. C. N. C. d. Azevedo, “Secure SIEM using OSINT for avoiding threats,” 2019. [Online]. Available: https://repositorio.ul.pt/bitstream/10451/31162/1/ulfc123948_tm_Jo%c3%a3o_Alves.pdf.

[24] ENISA, “Exploring the opportunities and limitations of current Threat Intelligence Platforms,” pp. https://www.enisa.europa.eu/publications/exploring-the-opportunities-and-limitations-of-current-threat-intelligence-platforms, 2017.

[25] C. Wagner, G. Wagener, A. Dulaunoy e A. Iklody, “MISP - The Design and Implementation of a Collaborative,” 2016. [Online]. Available: https://www.foo.be/papers/misp.pdf.

[26] NCI Agency, “Malware Information Sharing Platform,” 2017. [Online]. Available: https://academiamilitar.pt/images/site_images/Eventos/3rd_Conference/Day_1/MISP_usage_in_NATO_-_Johan_Schrooven.pdf.

[27] MISP Project, “MISP - Open Source Threat Intelligence Platform & Open Standards For Threat Information Sharing,” [Online]. Available: https://www.misp-project.org. [Acedido em 2019].

[28] MISP Project, “MISP Taxonomies,” [Online]. Available: https://www.misp-project.org/datamodels/#misp-taxonomies. [Accessed 2019].

[29] A. Cormack, X. Jansen, A. Moens e P. Peters, “Incident Classification / Incident Taxonomy according to eCSIRT.net - adapted,” 2015. [Online]. Available: https://www.trusted-introducer.org/Incident-Classification-Taxonomy.pdf.

[30] CIRCL.lu, “CIRCL Taxonomy - Schemes of Classification in Incident Response and Detection,” 2018. [Online]. Available: https://www.circl.lu/pub/taxonomy/. [Accessed December 2018].

[31] Microsoft, “Security intelligence,” [Online]. Available: https://docs.microsoft.com/en-us/windows/security/threat-protection/intelligence/. [Accessed December 2018].

[32] R. Azevedo, I. Medeiros e A. Bessani, “PURE: Generating Quality Threat Intelligence by Clustering and Correlating OSINT. In Proceedings of IEEE TrustCom, Rotorua, New Zealand.,” 2019. [Online]. Available: http://www.di.fc.ul.pt/~imedeiros/papers/TrustCom_2019_pure.pdf.

[33] R. Azevedo, I. Medeiros, G. Gonzalez-Granadillo, M. Faiella e S. Gonzalez-Zarzosa, “Enriching Threat Intelligence Platforms Capabilities,” 2019. [Online]. Available: http://www.di.fc.ul.pt/~imedeiros/papers/Secrypt2019_TEPT.pdf.

[34] F. Alves, A. Bettini, P. M. Ferreira e A. Bessani, “Processing Tweets for Cybersecurity Threat Awareness. Information Systems.,” 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306437920300727.

[35] M. de Bruijne, M. van Eeten, W. Pieters e C. H. Gañán, “Towards a new cyber threat actor typology: A Hybrid Method for the NCSC Cyber Security Assessment,” 2017. [Online]. Available: https://www.wodc.nl/binaries/2740_Summary_tcm28-273244.pdf.

[36] J. Mirkovic e P. L. Reiher, “A taxonomy of DDoS attack and DDoS defense,” p. https://www.researchgate.net/publication/2879658_A_taxonomy_of_DDoS_attack_and_DDoS_Defense_mechanisms, 2004.

[37] A. Saini, M. S. Gaur e V. Laxmi, “A Taxonomy of Browser Attacks,” 2015. [Online]. Available: https://www.researchgate.net/publication/290914540_A_taxonomy_of_browser_attacks.

[38] “VirusTotal,” Chronicle, [Online]. Available: https://www.virustotal.com/. [Accessed 2019].

[39] R. Vinot., “PyMISP,” [Online]. Available: https://pymisp.readthedocs.io/. [Accessed 2019].

[40] U.S. Joint Chiefs of Staff, “JP 2-0, Joint Intelligence,” 2013. [Online]. Available: https://www.jcs.mil/Portals/36/Documents/Doctrine/pubs/jp2_0.pdf.