Proposal No. 830929
Call H2020-SU-ICT-03-2018

Project start: February 1, 2019
Project duration: 42 months

Cyber Security for Europe

D3.3

Research challenges and requirements to manage digital evidence

Document Identification	
Due date	31 January 2020
Submission date	16 January 2020
Revision	1.0

Related WP	WP3	Dissemination Level	PU
Lead Participant	KUL	Lead Author	Davy Preuveneers (KUL)
Contributing Beneficiaries	UMU, UNITN, ATOS, C3P, CNR, DTU, KUL, POLITO, UMA	Related Deliverables	D3.1, D4.1 and D5.1

Abstract: This deliverable reports about the state-of-the-art, challenges, requirements and ongoing research within the frame of Task 3.4 on managing digital evidence as well as the reliability, safety and privacy guarantees of security intelligence techniques based on artificial intelligence, machine learning and data analytics. Based on the outcomes of deliverables '*D4.1: Requirements Analysis from Vertical Stakeholders*' and '*D5.1: Requirements Analysis of Demonstration Cases*', this report highlights relevant requirements and challenges on how to share digital evidence between different contemporary expert systems, on log and event management, on threat detection and security analytics with privacy-respecting big-data analytics. The goal of Task 3.4 is to describe the state-of-practice, identify the gap and research solutions to allow interoperability, either through the unification of languages, formats and interfaces, or through trusted intermediate translator systems respecting the privacy, business requirements and the regulations of the different countries. Hence, this report documents research tracks and assets with the ambition to bridge the gap with the state-of-the-art by addressing challenges and requirements for interacting with Threat Intelligence Information Services to capture evidence of malware activity at early stages, and to enable security intelligence in defensive systems by making sure the underpinning intelligence systems are fortified.

This document is issued within the CyberSec4Europe project. This project has received funding from the European Union's Horizon 2020 Programme under grant agreement no. 830929.

Executive Summary

This deliverable reports the results and outcomes of Task 3.4 on Security Intelligence. The goal of this task is to analyze and research new security intelligence and data analytics techniques to strengthen the security and privacy capabilities of cybersecurity applications in various vertical domains and use cases. The key topics addressed within this task and reported in this deliverable can be summarized as follows:

- Mechanisms to share digital evidence
- Threat intelligence information systems and services
- Interoperability in privacy, requirements and regulation
- Threat detection and security analytics
- Security intelligence in defensive systems

This report documents the relevant state-of-the-art in these areas as well as the gap to address the functional and non-functional requirements originating from the different vertical domains and use cases (UC) as documented in deliverables D4.1 and D5.1. Additionally, this document lists relevant components, algorithms, and software building blocks of the project partners that can help address these requirements. As these assets are at different levels of maturity, this document describes ongoing research tracks to address challenges and requirements to manage digital evidence:

- Lack of trust in the way threat intelligence information is handled by receiving parties is a key factor why organizations are reluctant to share information
- The quality (rather than the quantity) of threat feeds and events must increase for a reliable and automated threat analysis and mitigation
- The event based sharing philosophy of threat intelligence platforms does not match well with data driven and AI powered threat intelligence
- The application of security techniques – such as end-to-end encryption, onion routing, etc. – make it harder to harvest threat intelligence from monitoring data and event logs
- The AI capabilities of contemporary threat intelligence platforms enable new kinds of attacks that allow adversaries to learn how to evade detection
- Machine learning models that underpin cybersecurity solutions may leak sensitive information, and need strong protection to avoid privacy concerns or loss of reputation

These research challenges and requirements will be the main drivers to enhance existing assets and develop new ones within the frame of Task 3.4 to bridge the gap with the current state-of-practice and to increase the technological readiness towards a first set of demonstrators in WP5. A summary of the list of assets with the main highlights was previously reported as part of deliverable D3.1.

In this regard, this deliverable also describes a high-level architectural overview on how the different research activities and assets can be aligned into a common framework and global architecture that covers the different research activity and enabling cybersecurity technologies in Work Package (WP) 3. This architectural overview complements the generic architecture in D3.1 that represents all tasks in WP3, including Task 3.4 on Security Intelligence.

Document information

Contributors

Name	Partner
Davy Preuveneers	KUL
Giuseppe Manco	CNR
Massimo Guarascio	CNR
Susana Gonzalez Zarzosa	ATOS
Rolando Martins	C3P
Andrea Atzeni	POLITO
Jorge Bernal Bernabe	UMU
João Soares	C3P
Weizhi Meng	DTU
Roberto Doriguzzi Corin	UNITN/FBK
Ana Nieto	UMA
Alba Hita	UMU
Ivan Pashchenko	UNITN
Giorgio Di Tizio	UNITN
Daniele Canavese	POLITO

Reviewers

Name	Partner
Alireza Esfahani	UNILU
Welderufael B. Tesfay	GUF

History

0.01	2019-05-03	Davy Preuveneers	1st Draft
0.02	2019-05-22	Davy Preuveneers	Structure update
0.03	2019-05-11	Davy Preuveneers	Structure update
0.04	2019-07-08	Davy Preuveneers	Migration to OnlyOffice
0.05	2019-06-24	Giuseppe Manco	Add CNR asset
0.06	2019-06-25	Andrea Atzeni	Contributions by POLITO
0.07	2019-07-10	Davy Preuveneers	Structure update
0.08	2019-08-31	Davy Preuveneers	Merge input and revise structure
0.09	2019-09-13	Susana Gonzalez Zarzosa	Add ATOS asset
0.10	2019-09-19	Rolando Martins	Merging contributions C3P
0.11	2019-10-01	Davy Preuveneers	Major update to deliverable
0.12	2019-10-03	Giuseppe Manco	Updates to sections 5 and 6
0.13	2019-10-03	Massimo Guarascio	Updates to sections 5 and 6
0.14	2019-10-15	Weizhi Meng	Update on requirements
0.15	2019-10-15	Roberto Doriguzzi Corin	Update UNITN-FBK asset
0.16	2019-10-24	Jorge Bernal Bernabe	Update UMU asset
0.17	2019-10-25	João Soares	Update C3P assets

0.18	2019-10-30	Jorge Bernal Bernabe	Update UMU research
0.19	2019-11-04	Weizhi Meng	Update DTU research and asset description
0.20	2019-11-04	Roberto Doriguzzi Corin	Update UNITN-FBK asset
0.21	2019-11-04	Ana Nieto	Merge input UMA
0.22	2019-11-05	João Soares	Update C3P contribution
0.23	2019-11-06	Susana Gonzalez Zarzosa	Add ATOS updates
0.24	2019-11-15	Davy Preuveneers	Updates to SOTA
0.25	2019-11-21	Davy Preuveneers	Updates to Section 3
0.26	2019-12-01	Davy Preuveneers	Updates to Section 1
0.27	2019-12-06	Davy Preuveneers	Updates to Section 2
0.28	2019-12-06	Ivan Pashchenko, Giorgio Di Tizio	Updates to Sections 3, 5
0.29	2019-12-08	Weizhi Meng	Updates to Section 5 and typos
0.30	2019-12-10	Susana Gonzalez Zarzosa	Merge ATOS updates to Section 4 and 5
0.31	2019-12-14	Davy Preuveneers	Minor updates and fixes
0.90	2019-12-16	Davy Preuveneers	Fix spelling and layout issues
0.91	2019-12-27	Davy Preuveneers	Fix references to other deliverables
0.92	2019-12-31	Ivan Pashchenko	Updates to requirements mapping for RoCe
0.93	2020-01-02	Davy Preuveneers	Address review comments Alireza (UNILU)
0.94	2020-01-03	Davy Preuveneers	Address review comments Welde (GUF)
0.95	2020-01-03	Alba Hita	Updated Reliable-CTIs figure in Section 5
0.96	2020-01-03	Davy Preuveneers	Minor cleanups
0.97	2020-01-08	Davy Preuveneers	Update list of acronyms
1.00	2020-01-11	Davy Preuveneers	Final version

List of Contents

List of Figures

List of Tables

List of Acronyms

AES	Advanced Encryption Standard
AI	Artificial Intelligence
ABAC	Attribute Based Access Control
ANN	Artificial Neural Network
API	Application Programming Interface
APT	Advanced Persistent Threat
BA	Behavior Analytics
BGP	Border Gateway Protocol
CERT	Cyber Emergency Response Team
CIRCL	Computer Incident Response Center Luxembourg
CNN	Convolutional Neural Network
CP-ABE	Ciphertext-Policy Attribute-Based Encryption
CPS	Cyber Physical Systems
CPU	Central Processing Unit
CSIRT	Computer Security Incident Response Team
CT	Ciphertext
CTF	Capture The Flag
CTI	Cyber Threat Intelligence
CybOX	Cyber Observable eXpression
DDoS	Distributed Denial of Service
DFT	Digital Forensic Tools
DL	Deep Learning
DNN	Deep Neural Network
DNS	Domain Name System
DPI	Deep Packet Inspection
ENISA	European Network and Information Security Agency
EU	European Union
FTP	File Transfer Protocol
GDPR	General Data Protection Regulation
HIDS	Host Intrusion Detection System
HTTP	Hypertext Transfer Protocol
ICT	Information and Communications Technology
IdM	Identity Management
IdP	Identity Provider
IDS	Intrusion Detection System
IoC	Indicator of Compromise
IP	Internet Protocol
IPS	Intrusion Prevention System
IRC	Internet Relay Chat
ISHA	Information Sharing and Analytics
IT	Information Technology

JSON	JavaScript Object Notation
MISP	Malware Information Sharing Platform
ML	Machine Learning
MLaaS	Machine Learning as a Service
NIDS	Network Intrusion Detection System
NFV	Network Function Virtualisation
OT	Operational Technology
OSINT	Open Source Intelligence
PCAP	Packet Capture
PET	Privacy Enhancing Technologies
PSD2	Payment Service Directive 2
RBAC	Role Based Access Control
REST	Representational State Transfer
SDN	Software Defined Networking
SIEM	Security Information and Event Monitoring
SMTP	Simple Mail Transfer Protocol
SOC	Security Operations Center
SP	Service Provider
SQL	Structured Query Language
SSH	Secure Shell
SSL	Secure Sockets Layer
STIX	Structured Threat Information eXpression
TAXII	Trust Automated eXchange of Indicator Information
TCP	Transmission Control Protocol
TIP	Trust Intelligence Platform
TLP	Traffic Light Protocol
TOR	The Onion Router
UC	Use Case
UMA	User Managed Access
URL	Uniform Resource Locator
VPN	Virtual Private Network
WP	Work Package

1 Introduction

Advances in distributed computing, storage and communication technologies, the availability of growing amounts of data, as well as the adoption of new business models, have been major drivers behind key computing paradigms, such as cloud and mobile edge computing, the Internet of Things (IoT) and Cyber Physical Systems (CPS), Big Data analytics and Artificial Intelligence (AI), and 5G. These technology paradigm shifts and the ongoing digital transformation within vertical domains – such as healthcare, manufacturing and Industry 4.0, e-commerce, finance and banking, smart homes and cities, connected vehicles, and beyond – have not only resulted in an increased interconnectivity within the boundaries of a single home or organization, but also across those boundaries, such as in supply chains of collaborating production companies and enterprises.

The downside of these emerging trends is that interconnected ICT systems increase the attack surface of critical infrastructures [Ten 2010], and they have therefore become a valuable target for malicious adversaries and cybercriminals that aim to disrupt services to exfiltrate sensitive data, or to abuse the victim machines and networks for performing other malicious activities. Indeed, the accelerating wave of sophisticated attacks and advanced persistent threats is a growing security and privacy concern for both the consumer and businesses. In their 2019 Data Breach Investigations Report [Verizon 2019], Verizon acknowledged that ransomware remains among the most popular and pervasive malware variants. Cybersecurity firm Emsisoft documented in their report [Emsisoft 2019] how 621 hospitals and 500 schools and 169 businesses were attacked by ransomware. More recently, ransomware was creating havoc in hospitals in France[1], the US and Australia[2], limiting the use of their computer systems and even causing the loss of patient data. The manufacturing world has also not been spared from these cyber threats. The devastation of the NotPetya malware – designed to destroy an organization's capability to process data – on US-based pharmaceutical giant Merck[3] in 2017 cost the company $300 million in lost revenue. More recent EU cases of ransomware attacks against manufacturers are those at the automation tool manufacturer Pilz in Germany[4], aircraft parts manufacturer Asco in Belgium[5], Norwegian aluminum producer and manufacturer Norks Hydro[6], etc. Even if a company has tight security policies and cyber defenses in place, the collaboration with external suppliers remains a security challenge. This was recently demonstrated in the intellectual property theft case at Airbus[7]. The company was attacked through weak links in the Airbus supply chain, in this case the Virtual Private Network (VPN) that supplier's employees use to connect to Airbus.

As the impact of a breach is not always isolated to within a single organization or country, it is crucial that organizations not only develop perimeter defenses to keep attackers out, but also invest in cyber threat

[1] https://www.infosecurity-magazine.com/news/french-hospital-crippled-by/
[2] https://nakedsecurity.sophos.com/2019/10/03/ransomware-attacks-paralyze-and-sometimes-crush-hospitals/
[3] https://www.cyberscoop.com/notpetya-ransomware-cost-merck-310-million/
[4] https://www.zdnet.com/article/major-german-manufacturer-still-down-a-week-after-getting-hit-by-ransomware/
[5] https://www.cybersecurity-insiders.com/aircraft-parts-manufacturer-asco-hit-by-a-ransomware-attack/
[6] https://www.securityweek.com/ransomware-attack-costs-norsk-hydro-tens-millions-dollars
[7] https://www.techradar.com/news/airbus-hacked-through-supplier-vpns

intelligence inside that perimeter to monitor corporate networks so that they (1) can detect how and when a breach occurs, (2) are able to identify compromised systems, (3) can determine how adversaries modified their systems or identify which data was stolen, (4) contain the incident from further contamination, and (5) are able to remediate these incidents and recover from the breach. To prevent the same incident from happening elsewhere, enabling technologies are needed to manage digital evidence, i.e. techniques to collect, examine, analyze and possibly share digital evidence originating from a variety of digital data sources. Nonetheless, there are limitations with contemporary solutions to adequately address and mitigate cyber threats in a timely manner.

The amount, sophistication and impact of cyber-attacks is increasing and so is the amount of protected end-points and the volumes of data coming from these end-points that need to be scrutinized for anomalous behavior. Traditional solutions, such as Security Information and Event and Management (SIEM) solutions and Intrusion Prevention Systems (IPS), may lack the necessary capabilities to quickly adapt to new and/or evolving threats. To address this concern and reduce the time to mitigation in case of an attack, contemporary cyber threat detection solutions need to become more intelligent in order to automate detection and response in an effective manner. Such solutions would need to process and audit large amounts of monitoring data and event logs to identify and automatically react to suspicious activities. Given the increasing heterogeneity and volume of data, the importance of machine learning (ML) and Big Data analytics in the area of security and cyber threat intelligence is growing to handle this data [Mahmood 2013]. At the same time, these AI solutions have been demonstrated to be subject to their own category of threats [McDaniel 2016]. New threats are able to fool classification or anomaly detection solutions. For example, small perturbations on the input data, i.e. adversarial inputs, are sufficient for a spoofing or evasion attack against a defensive system if the underlying ML models were not trained in a robust manner.

To enhance detection and prevention of cyber threats, organizations collaborate to define defensive actions against complex attack vectors by sharing information and knowledge about threats, sightings, indicators of compromise (IoC), and mitigation strategies. This means interoperability across different security expert systems becomes a necessity. Threat Intelligence Platforms (TIP) have therefore become a critical security component within the enterprise to deal with the increasing volume and sophistication of cyber-attacks [Tounsi 2018]. These software platforms are cloud or on-premise systems that facilitate the aggregation and correlation of threat events from different parties and multiple sources [Qamar 2017], including security monitoring and data analytics tools. To simplify the sharing of threat information between different TIPs, they often rely on standardized data exchange formats, such as Structured Threat Information eXpression (STIX) 2.0[8]. Other messaging and data formats to describe threat properties and attributes are Trust Automated eXchange of Indicator Information (TAXII) [9], and Cyber Observable eXpression (CybOX) [10]. While such data formats have emerged for better interoperability across signature based detection systems, they may not be ideally suited for today's AI powered threat detection systems. Indeed, the sharing of

[8] http://docs.oasis-open.org/cti/stix/v2.0/stix-v2.0-part2-stix-objects.html
[9] https://taxiiproject.github.io
[10] https://cyboxproject.github.io

Indicators of Compromise (IoC) is incompatible with data-driven machine learning approaches incorporated in advanced monitoring and detection products, and even combining all the threat intelligence feeds of all providers may lead to incomplete intelligence, as already confirmed in the 2015 Data Breach Investigations Report of Verizon.

Beyond the technical challenges to realize interoperability, trust remains a key challenge in collective cyber threat intelligence (CTI), especially when the incident data collected and shared in TIPs may harm the reputation of a reporting organization, or when the data is used to automatically trigger cybersecurity solutions. Reporting organizations need to trust the receiving organizations that the information is handled confidentially, but at the same time do the receiving organizations assume that the information itself is reliable and trustworthy. For example, TIPs allow to export a subset of the data into Intrusion Detection System (IDS) rules that can be inserted into software solutions such as Snort[11] or Suricata[12]. Obviously, malicious or unreliable input information may compromise the usefulness of such Host Intrusion Detection Systems (HIDS) or Network Intrusion Detection Systems (NIDS) or may even harm an innocent targeted organization. The assumption is that the information collected within TIPs is reliable and accurate. This may not be the cause, either because of noise in the monitored data, inaccurate detection patterns, or due to malicious intent.

Additionally, as TIPs may collect and process sensitive information, care must be taken to ensure that the management of digital evidence and the mandatory notification of cyber incidents are compliant with relevant regulations and directives, such as the EU General Data Protection Regulation (GDPR) or the Payment Service Directive 2 (PSD2), each with their own timings and criteria for notification.

1.1 General objective

The general objectives for an adequate security and cyber threat intelligence are (1) interoperability for effective and trustworthy threat intelligence sharing, (2) security intelligence to close the loop with automated detection, prevention and response, (3) privacy-respecting security analytics to remain compliant with relevant regulations and directives, and (4) robustness such that intelligent solutions to enhance security do not increase the attack surface themselves.

The goal of ‘Task 3.4 Security Intelligence’ in WP3 is to analyze and research new security intelligence and data analytics techniques to strengthen the security and privacy capabilities of cybersecurity applications in various vertical domains and use cases. The key topics addressed within this task can be summarized as follows:

- Mechanisms to share digital evidence
- Threat intelligence information systems and services
- Interoperability in privacy, requirements and regulation
- Threat detection and security analytics

[11] https://www.snort.org
[12] https://suricata-ids.org

- Security intelligence in defensive systems

This report documents relevant state-of-the-art in these areas (although by no means with the ambition to be exhaustive with a complete literature review) as well as the gap to address the functional and non-functional requirements originating from the different vertical domains and use cases (UC) as documented in deliverables D4.1 [D41 2019] and D5.1 [D51 2019].

Additionally, this document lists relevant components, algorithms, and software building blocks of the project partners that can help address these requirements. As these assets are at different levels of maturity, this document describes ongoing research tracks to address challenges and requirements to manage digital evidence. These research challenges and requirements will be the main drivers to enhance existing assets and develop new ones within the frame of Task 3.4 to bridge the gap with the current state-of-practice and to increase the technological readiness towards a first set of demonstrators. A summary of the list of assets with the main highlights was previously reported as part of deliverable D3.1 [D31 2019].

1.2 Participants

This document contains contributions of a mix of research and industrial partners covering a broad spectrum of security intelligence. The deliverable reports mainly about the activities carried out in T3.4, led by KU Leuven, in which the following project partners are involved:

-	Universidad de Murcia	Spain
-	Università degli Studi di Trento	Italy
-	ATOS	Spain
-	Universidade do Porto	Portugal
-	Consiglio Nazionale delle Ricerche	Italy
-	Danmarks Tekniske Universitet	Denmark
-	KU Leuven	Belgium
-	Politecnico di Torino	Italy
-	Universidad de Málaga	Spain

These partners bring complementary expertise to the table, as well as various assets that help address the above cybersecurity goals and bridge the gap with the state-of-the-art with a unified security architecture that spans the different tasks in WP3.

1.3 Main outcomes and structure of this deliverable

The main contributions and structure of this deliverable can be summarized as follows:

- *Section 2:* Review of relevant related work in the area of security intelligence and the management of digital evidence.
- *Section 3:* Overview of the research activities of the project partners in T3.4.
- *Section 4:* Identification of key security intelligence requirements across different vertical domains based on the input of deliverables D5.1 and D4.1.
- *Section 5:* An overview of assets (algorithms, methods, libraries, platforms) contributed by the partners in the frame of T3.4 and at different levels of maturity.

- *Section 6:* The mapping of these requirements onto the assets provided, and identification of gaps and directions for further research.
- *Section 7:* Concludes with a summary of research challenges and requirements to manage digital evidence.

2 State-of-the-art

This section provides a high-level overview of the state-of-the-art on security intelligence from both a human and system perspective. We will discuss previous research covering the inputs, the processing, and the sharing of digital evidence, the actionable insights/outcomes for threat intelligence, and implications from a privacy point-of-view.

2.1 Cyber threat intelligence

Threat intelligence [Bromiley 2016] is any evidence-based knowledge - obtained from human and/or technical sources - about threats that can inform decisions [McMillan 2013], with the aim to prevent an attack or shorten the window between compromise and detection. Cyber-attacks are growing in sophistication, and are demanding for novel cyber security defenses. Tounsi et al. [Tounsi 2018] reviewed various trends in this landscape and how static approaches using signature schemes and heuristics based approaches no longer suffice to deal with the dynamic nature of contemporary threats and their capabilities to evade detection. Indeed, new cyber-attacks are multi-vectored (multiple means of propagation) and multi-staged (e.g. infect and spread across the network in multiple stages to exfiltrate data), and they can be polymorphic or personalized making them harder to detect with signature based approaches. In their survey, the authors acknowledge the need for organizations to share cyber threat information and to transform that information into threat intelligence to prevent attacks or to minimize disaster recovery efforts. The authors review emerging research, ongoing trends, useful standards and compare selected tools (including MISP, CRITs, Soltra Edge, CIF v3, Threatelligence, AlliaCERT TI tool). They also confirm the reluctance of organizations to share threat intelligence, as well as reasons behind this observation, as already discussed earlier. The authors propose to research solutions for sharing threat intelligence based on trust and anonymity to mitigate a.o. the risks of leaks of sensitive business information, and to reduce the large amount of threat feeds as they significantly overlap in content or lack sufficient contextual information. This information may enable ML models to come up with new ways to detect malwares based on what they have in common with others, rather than with a 1-for-1 approach where each threat is mapped to a signature or/and IoC. Last but not least, their survey confirms the need for a standardized representation of threat intelligence information to improve the quality of threat intelligence, hereby providing better support for automated analytics solutions on large volumes of data. The same shift towards AI and ML based approaches was also confirmed as an opportunity by Conti et al. [Conti 2018] in the introductory chapter of their book on `Cyber Threat Intelligence'.

2.1.1 Intrusion detection

Network-based Intrusion Detection System (NIDS) monitors and logs all the packets inside a network, searching for known events. These systems are capable of disassembling packets, parsing headers and payload data and identifying protocols correctly without taking the source and destination port into account. Most of them are able to block intrusions by identifying signatures in the traffic and then rejecting or allowing packets according to certain rules.

In this section, we cover three well-known systems of this type: Snort, Suricata and Bro. Those are excellent solutions for detecting intrusions and complex attacks in networks.

Snort

Snort [Roesch 1999, Snort 2017] is a leading free and open source network intrusion detection system (NIDS) and intrusion prevention system (IPS). It is more than two decades old and currently maintained by Cisco. It analyzes Internet Protocol (IP) traffic in real time and can store log files for post mortem analysis. Snort offers a substantial ruleset to monitor traffic, and this ruleset is regularly updated by the community as well as commercial enterprises. When the traffic matches one of its rules, appropriate actions are taken (alert, drop, etc.). Snort offers pre-processors to reshape the traffic into a useful format before matching with its rules. Furthermore, Snort can be made application aware – i.e. identify the applications that generated the traffic – by using detectors operating at layer 7 of the network stack.

One of the limitations of Snort (version 2.x) is its single threaded design, with only one core being used irrespective of how many cores a Central Processing Unit (CPU) provides. This is one of the reasons why the performance of the solution was compared a.o. with Suricata [Park 2017] in the frame of contemporary network infrastructures. Over the years, there has been an increase in network traffic, which in turn increases the processing demands on IDS infrastructure, especially when carrying out deep packet inspection (DPI). Alternatives like Suricata support multithreading out of the box. While workarounds for Snort exist (e.g. running multiple Snort 2.x instances on different cores), version 3.x of Snort will properly address this performance concern by supporting multiple packet processing threads.

Suricata

Suricata is a well-known open source IDPS [Suricata 2016]. In addition to real-time Network security monitor (NSM), intrusion detection and offline Packet Capture (PCAP) processing, it implements inline intrusion prevention, i.e., it is capable of both detecting and preventing known attacks using different mechanisms such as anomaly detection or signature-based filtering, dropping compromised packets before they reach their destination.

Suricata inspects network packets according to certain rules and signature checking procedures but it also took a step beyond these simple approaches by supporting LUA scripting in order to detect complex intrusions by defining multiple rules [Schreiber 2016]. Suricata also moved up in the Open Systems Interconnection (OSI) model, since it is capable of analyzing and interpreting application layer traffic. This new feature can be used to perform advanced Hypertext Transfer Protocol (HTTP) processing or even detect web server intrusions. It does not just log packets, it is possible to extract potentially malicious executables, Secure Sockets Layer (SSL) certificates, requests or queries, file signatures and other objects by automatically detecting protocols, regardless of the destination port [Stanger 2015]. Among other features, it supports flow tracking, both IPv4 and IPv6, GeoIP, IP reputation and Domain Name System (DNS) parsing [OISF 2015].

Suricata is also known for its efficiency and performance since it is a multithreaded solution and takes advantage of hardware acceleration [Schreiber 2016]. According to White, Fitzsimmons and Matthews, Suricata is better than Snort in terms of performance [White 2013]. The performance of both systems was compared while scaling the number of CPU cores and varying configurations such as using multi-instance Snort. According to Fekolkin [Fekolkin 2015], the precision of the rules used by Suricata and Snort affects the rate of false negatives and false positives in terms of threat detection [Bro 2017]. These rules can trigger

actions, just like Snort, after the packet analysis phase. It is also important to mention that Suricata was mostly based on Snort, but besides being more modern, it took a step further into achieving a greater scalability, efficiency and precision.

Suricata can detect malware by analyzing signatures of executables or parts of binaries and then searches for known signatures, which is sometimes enough to detect common malware. However, it is a difficult task to detect new malware or new attacks, even for anti-viruses, which also use signature-based techniques among other methods. Regardless of this limitation, the possibility of saving malicious executables and other payloads present in the captured traffic for post-processing is also a plus while studying malware and what kind of attacks are being used against a company or organization.

Bro IDS

Bro is an open source framework for network analysis and security monitoring, which can be used to build a powerful NIDS for UNIX systems. It was created by Vern Paxson in 1995 and it has been developed by researchers and students of the International Computer Science Institute (ICSI) [Bro 2017]. It is a passive traffic analyzer which uses DPI to detect intrusions in the network traffic. It is also capable of performing tasks not related to security, such as traffic baselining and performance tests. There are many built-in functionalities: Bro parses application data, extracts files from Transmission Control Protocol (TCP) streams, identifies outdated versions of software, logs every event in the network in a well-structured format, which can be externally analyzed by other software and stored in databases, and much more. However, users can write their own scripts in order to achieve complex tasks, using a domain-specific, Turing-complete scripting language with an event-based programming model [Bro 2017].

Unlike Snort and Suricata, which are capable of performing intrusion prevention, Bro does not work in inline mode [Khalil 2015]. It is a policy based IDS that generates logs and also an excellent solution for intelligence gathering. It uses *libpcap* in order to filter the captured packets at the kernel level and reduce the workload by selecting packets needed by the current policy [Sommer 2003].

Bro has two main components: the event engine and the policy script interpreter. Packets are processed by the event engine, which performs processing task, such as state management and protocol parsing, and generates a stream of events which are passed to the policy layer [36]. The policy script processes these events with the scripts supplied by users. Users can define event handlers in these scripts and therefore perform certain actions, working with a high-level abstraction of the packets. These actions can even launch user-supplied scripts or external programs in order to trigger an active response to an attack [Bro 2017].

In terms of scalability, Bro supports clustering for large-scale deployments and is also capable of performing both offline and real-time analysis. It supports many application layer protocol analysis, such as DNS, File Transfer Protocol (FTP), HTTP, Internet Relay Chat (IRC), Simple Mail Transfer Protocol (SMTP), Secure Shell (SSH) and SSL, and also non-application layer analysis, which include built-in analyzers that can detect port scanning techniques. It also supports IPv6 and tunnel detection and further tunnel traffic analysis using decapsulation techniques. Bro supports alternative backends, such as Elasticsearch, Logstash, and Kibana by Elastic [Elastic 2016], since all connections, sessions, and application level data are written to a large set of log files [Smith 2017], which can be useful to normalize and analyze Bro logs. Signature detection techniques are also supported and Snort rules can also be easily imported.

2.1.2 Machine learning powered threat intelligence

Traditional static cyber threat solutions rely on signature schemes or rules to detect attacks. Such schemes are known to have difficulty with defending against unknown attacks. That is why AI and ML, including deep learning techniques, are being adopted to handle these concerns.

Homayoun et al. [Homayoun 2018] applied deep learning on network traffic to detect botnets. They argue that existing intrusion detection systems are unlikely to be effective at countering advanced techniques (such as encrypted payloads) deployed in recent botnets. That is why they propose BoTShark as a solution to analyze traffic without relying on deep packet inspection. Their solution compares the effectiveness of Autoencoders and Convolutional Neural Networks (CNN) for botnet detection. The authors consider the application of other deep learning techniques such as Long Short Term Memory (LSTM) as future work.

Similar work was presented by Ding et al. [Ding 2018]. Their approach also relies on machine learning techniques, but with a particular focus on detecting anomalies in network traffic and more particularly in the Border Gateway Protocol (BGP). The authors acknowledge that detecting network anomalies and intrusions are crucial to counter cyber-attacks and safeguard the security of service providers and network customers. A key concern for security research is the difficulty to distinguish anomalies due to failures and malicious intent. In their work, the authors consider worms (such as Slammer, Nimda, and Code Red I worms), power outages, and BGP router configuration errors as examples of anomalous events. They discuss relevant datasets and useful high-level features to aid with the detection.

We will not discuss how machine learning and deep learning has been applied in the area of cybersecurity and in different security use cases. For interested readers, we refer to the following surveys and literature reviews [Buczak 2015, Gardiner 2016, Jiang 2016, Mahdavifar 2019].

2.1.3 Reliability, safety and privacy guarantees of threat intelligence techniques

While advances have been made in the past few years in the domain of Big Data for security analytics and advanced machine learning techniques - including deep learning and data/pattern mining - several key challenges remain for which the state-of-the-art is yet to produce an adequate answer. The machine learning (ML) algorithms used in security analytics applications are also subject to attacks [Barreno 2010]. As the interest and use of Machine Learning for security applications increases, so too will the awareness from cyber-criminals. When frequently updating an ML model to account for new threats, malicious adversaries can launch causative/data poisoning attacks to intentionally inject misleading training data so that an ML model becomes ineffective. For example, various poisoning attacks on specific ML algorithms [Rubinstein 2009, Biggio 2012] were able to bypass intrusion detection systems. More recently, Chen et al. [Chen 2018] demonstrated how to automate poisoning attacks against malware detection systems. However, keeping ML models fixed can give rise to exploratory/evasion attacks to find the blind spots. Understanding the trade-offs between keeping a machine learning model fixed versus updating the model fixed versus the complexity of the model is non-trivial.

Machine learning models are becoming commercially valuable assets and from a business perspective the confidentiality of these models needs to be protected. Another reason from a security point of view for keeping models confidential is that malicious adversaries may use stolen models to help evade detection. However, various works have demonstrated that it is feasible to steal ML models when they are exposed

through APIs that are publicly accessible query interfaces for prediction or classification. For example, Tramèr et al. [Tramèr 2016] demonstrated the feasibility of model stealing attacks against the online services of BigML and Amazon Machine Learning. An obvious countermeasure against such attacks is restricting the information provided by ML APIs. However, the authors acknowledge that attackers may adaptively choose inputs based on only class labels and launch retraining attacks. Depending on the amount of input samples and labels, an attacker can construct a machine learning with similar prediction capabilities and exploit the transferability property of adversarial examples. Such a retraining attack can be mitigated by imposing a cap on the usage of the ML APIs, also known as a query budget. Kesarwani et al. [Kesarwani 2018] proposed a solution that monitors and quantifies the extraction status of a model by observing the API query and response streams. They present and empirically evaluate two strategies that are based on respectively information gain and feature space coverage to estimate the learning rate not only of a single user but also of colluding adversaries. The feasibility of their proposed approach was demonstrated on decision tree and neural network models, open source datasets and the BigML MLaaS platform.

Privacy is also a concern from a machine learning point of view. Indeed, machine learning models are trained on large amounts of data. The fact that someone may be able to infer whether an individual's sensitive data was used to train a model may constitute a privacy threat, especially if such a machine learning model is offered as a service (MLaaS). Nasr et al. [Nasr 2018] indeed investigated solutions to address the fact that machine learning models can leak a significant amount of information about training sets and predictions. These threats are known as (black-box) membership inference attacks. They propose a mechanism to train models with membership privacy. Their solution ensures indistinguishability between the predictions of a model on training data and other data points from the same distribution. They show with 3 datasets, including CIFAR100, Purchase100 and Texas100, that they can mitigate membership inference attacks – i.e. comparable to an adversary doing random guesses – with a negligible drop of less than 4% in the model's prediction accuracy.

2.2 Managing and sharing threat intelligence

Nowadays, organizations collaborate [Turner 2016] to define defensive actions against complex attack vectors by sharing information and knowledge about threats, sightings, indicators of compromise (IoC), and mitigation strategies. Threat Intelligence Platforms (TIP) have therefore become a critical security component within the enterprise to deal with the increasing volume and sophistication of cyber-attacks. These software platforms are cloud or on-premise systems that facilitate the aggregation and correlation of threat events from different parties and multiple sources, including host and network security monitoring systems and AI powered data analytics tools. Burger at al. [Burger 2014] provide a taxonomy for cyberthreat intelligence information exchange technologies.

2.2.1 Enabling technologies for sharing threat intelligence

Wagner et al. [Wagner 2016] presented MISP, an open source threat intelligence sharing platform, initially focusing on malware information, but now also used for other threat vectors, such as financial indicators for fraud detection and prevention. MISP operates on events and attributes, as depicted in Figure 1. Events typically encapsulate tags to link events with one another, objects from other information sharing tools, and attributes with various system or network related indicators. The category of an attribute puts it in a certain context, whereas the type of an attribute describes the indicator.

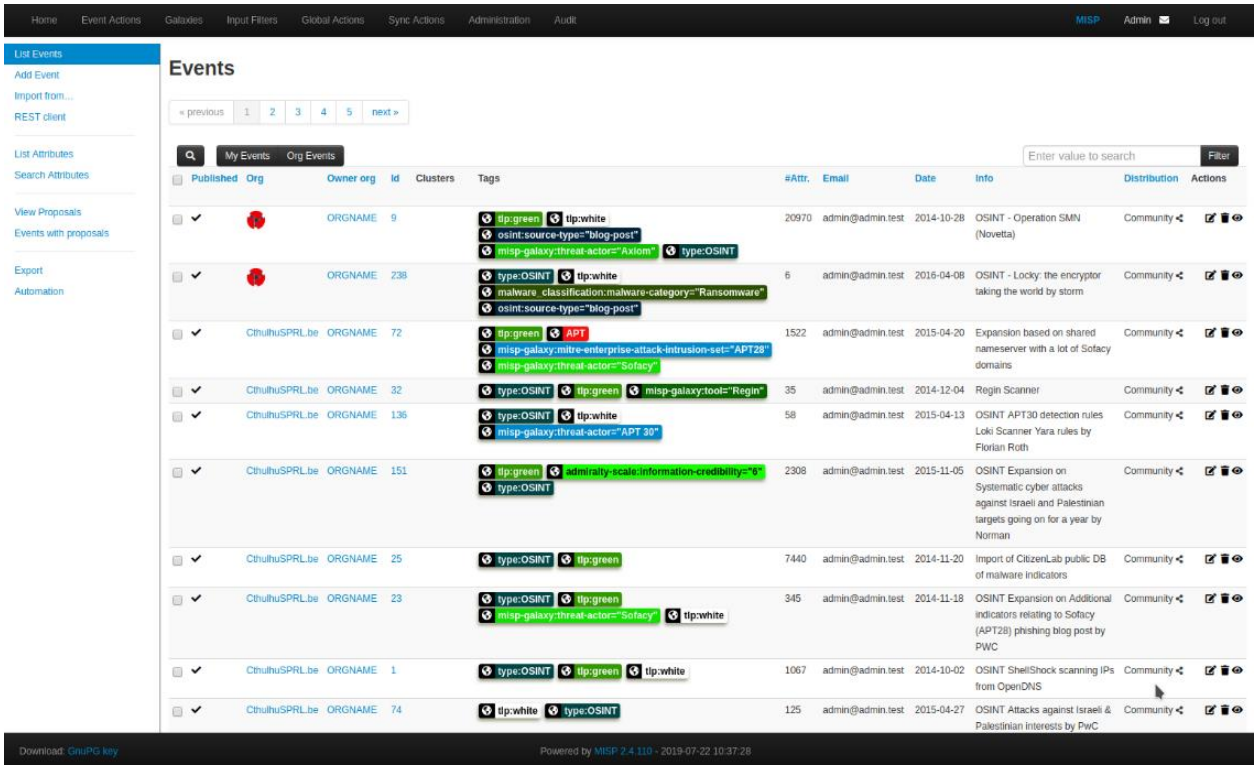

Figure 1: Example of a MISP instance

A typical example of a (relatively short) threat event of the CIRCL OSINT threat intelligence feed[13], including tags and attributes, in MISP's JSON format is depicted below in Figure 2.


```
{
    "Event": {
        "info": "test enforcewarningslists",
        "Tag": [
            {
                "colour": "#339900",
                "exportable": true,
                "name": "tlp:green"
            },
            {
                "colour": "#82b37e",
                "exportable": true,
                "name": " Decoy"
            }
        ],
        "publish_timestamp": "0",
        "timestamp": "1559842491",
        "analysis": "0",
        "extends_uuid": "",
```


[13] https://www.circl.lu/doc/misp/feed-osint

```
            "published": false,
            "date": "2019-06-06",
            "Orgc": {
                "uuid": "55f6ea5e-2c60-40e5-964f-47a8950d210f",
                "name": "CIRCL"
            },
            "threat_level_id": "3",
            "uuid": "5cf94d9e-cdfc-462d-9f1c-4e18950d210f"
        }
}
```


Figure 2: Example of a threat event in MISP JSON format

To simplify the sharing of threat information between different TIPs, they often rely on standardized data exchange formats, such as STIX 2.0. In previous work, Menges et al. [Menges 2018] define a general model for incident reporting and review state-of-the-art incident reporting formats – including STIX, IODEF, VERIS, and X-ARF - against this model. They analyse these formats for their strengths and weaknesses in light of suitable use cases.

Due to the sensitive nature of the threat events that TIPs collect, the way the information is shared with cyber emergency teams and other security stakeholders is subject to access rules that designate who is authorized to receive certain information. The Traffic Light Protocol (TLP)[14] is a well-known scheme developed to facilitate and encourage the exchange of threat events [ENISA 2019]. The user sending an event to the platform (i.e. the producer) assigns it a color that indicates the appropriate audience with whom it may be further disseminated (i.e. the consumer). Information within a threat event may be intended:

- TLP:RED: Only for the direct addressees (e.g. those present at a meeting)
- TLP:AMBER: For (certain people within) an organization
- TLP:GREEN: For a community (e.g. peers and partner organizations)
- TLP:WHITE: To be freely disseminated (but subject to copyright rules)

Sauerwein et al. [Sauerwein 2017] conducted a systematic study of 22 threat intelligence sharing platforms, including MISP. A comparison of these platforms resulted in eight key findings, including (1) the lack of a common definition of threat intelligence sharing platforms, (2) STIX being the de-facto standard for threat information, (3) the sharing of indicators of compromise as main goal, (4) the closed source nature of most platforms, (5) the focus on data collection rather than analysis, (6) neglected trust issues, (7) an increasing interest in academia and industry, and (8) manual tasks making the user a bottleneck. In their report [ENISA 2018], ENISA also confirmed TIPs do not only offer opportunities, but have trust limitations:

1. The event producer trusts the platform provider to not expose confidential data to unauthorized recipients.
2. The event producer trusts the event consumers that they handle shared information according a predefined protocol (e.g. TLP).

[14] https://www.us-cert.gov/tlp

3. The platform provider and event consumers trust the event producer that the information shared is reliable and credible.

We refer to [Sauerwein 2017, ENISA 2018, Tounsi 2018, Chantzios 2019] for more details on the typical functionalities and sharing capabilities of threat intelligence sharing platforms.

2.2.2 Privacy and reputation challenges

Indicators of compromise and threat sightings may carry sensitive confidential information, and affected parties may be reluctant to share this threat information with other security stakeholders. Without appropriate measures, parties may only be willing to share intelligence with those parties with whom they already established a trust relationship. This concern was explored by van de Kamp et al. [van de Kamp 2015]. The authors propose cryptographic approaches to hide details of an indicator or sighting while still enable sharing, hereby limiting the possibility of information misuse. The proposed method implemented on top of MISP relies on hashing with non-secret salts chosen at random for each IoC such that precomputation of the hashes is not possible. While technically feasible, it limits the data analysis and correlation of related events. Furthermore, the values of certain typed attributes may not meet certain formatting criteria (e.g. a hostname or IP address), which may limit the practical feasibility of the hashing method as a privacy enhancing technology.

2.3 Privacy-respecting Big Data analytics

The previous sections already highlighted various privacy concerns in the area of machine learning and in threat intelligence platforms. In this section, we will highlight complementary concerns related to the analysis of threat intelligence information through Big Data analytics solutions.

2.3.1 Privacy and compliance implications

Managing digital evidence of an incident in an inherently distributed system is often the result of painstakingly analyzing a multitude event logs. Big Data platforms that can digest a variety of event streams in (near) real-time may offer a solution. For example, existing Big Data frameworks, such as Apache Spark 2[15], offer solutions for distributed or federated machine learning on large data sets. However, they are not equipped to deal with strong security and privacy compliance constraints where, for example, certain sensitive data should never leave a particular machine for further processing and analysis. Also, with different adversarial settings and performance trade-offs for model vs. data pallarellization, there is no single machine learning-based solution (i.e. ML algorithm, model, set of parameters, etc.) that fits all security needs of every business. Health and finance specific solutions may need to address additional stringent confidentiality requirements (e.g. the HIPAA[16] and HITECH[17] industry standards for health companies, the Privacy of Consumer Financial Information Rule[18] for financial institutions). Managed service oriented security solutions may face specific real-time performance and scalability concerns to detect anomalies in

[15] https://spark.apache.org
[16] https://www.hhs.gov/hipaa/for-professionals/security/laws-regulations/index.html
[17] https://www.hhs.gov/hipaa/for-professionals/special-topics/hitech-act-enforcement-interim-final-rule/index.html
[18] https://www.ftc.gov/tips-advice/business-center/guidance/how-comply-privacy-consumer-financial-information-rule-gramm

large amounts of security events from different end-points. These domain and application specific security needs impose unique challenges on the underlying data analytics and machine learning building blocks.

In previous work, Kantarcioglu et al. [Kantarcioglu 2019] discussed a variety of research challenges at the intersection of Big Data, security and privacy. Key aspects that were highlighted related to the storage and the querying of data, the linking and sharing of data, the analysis of big data, accountability challenges in big data, the role of blockchains, as well as the implications of adversarial machine learning for cybersecurity. Gruschka et al. [Gruschka 2018] reviewed privacy issues and data protection challenges in Big Data by means of a case study analysis, and this in the frame of the General Data Protection Regulation (GDPR). A similar review of security and privacy challenges in Big Data and roadmaps for the future were identified in [Nelson 2017, Altman 2018, Bao 2018].

2.3.2 Privacy enhancing technologies for Big Data

With the increased attention on privacy, many solutions are being proposed in the literature, including deidentification software solutions such as ARX[19] [Prasser 2015] under a variety of privacy models (including k-anonymity, l-diversity, t-closeness, and differential privacy). From a machine learning perspective, there is ongoing research on privacy-preserving machine learning [Abadi 2016, Bonawitz 2017] and the application (and limitations) of federated machine learning [Briland 2017], possibly complemented with cryptographic building blocks (such as security multi-party computation [Laud 2015] and homomorphic encryption [Phong 2018]) to learn a (shared) model that maintains its prediction or classification capabilities while offering certain privacy guarantees.

[19] https://arx.deidentifier.org/

3 Research activities

In this section, we highlight the main research activities of the project partners within T3.4, and cluster the contributions along 5 lines of research. These topics will be consolidated in an security intelligence platform, as depicted below, that will be part of an overarching architecture defined at the level of WP3, and documented with more details in deliverable D3.1 [D31 2019].

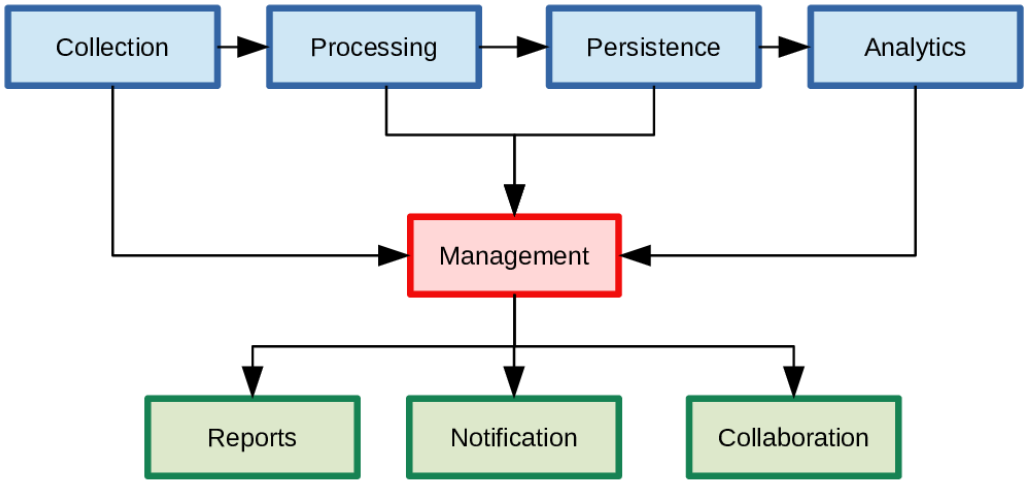

Figure 3: High-level overview of a collaborative security intelligence platform

The above Figure 3 breaks down the individual steps in a security intelligence platform, from collecting and aggregating data from system monitors, to AI based analytics, the creation of management reports, the mandatory notification to legal or supervisory authorities, and the sharing of threat intelligence with collaborating organizations.

3.1 Expected research activities per partner

The research activities of the project partners are discussed along the following main themes, and summarized in the table at the end of this section:

- Mechanisms to share digital evidence
- Threat intelligence information systems and services
- Interoperability in privacy, requirements and regulation
- Threat detection and security analytics
- Security intelligence in defensive systems

Many times the term threat intelligence is used only to make reference to cybersecurity related information about threats or vulnerabilities that is exchanged through sharing platforms such as MISP. However,

considering for example the definition of Threat Intelligence provided by the SANS Institute [Bro16] or the one provided by ThreatConnect [Thr18] ("*TI is the knowledge of a threat's capabilities, infrastructure, motives, goal and resources. The application of this information assists in the operational and strategic defense of network-based assets*", organizations should consider not only cybersecurity information coming from external sources or data feeds but also from internal ones. This means that information collected using security tools deployed in the own monitored infrastructure, such as firewalls, Security Information and Event Management (SIEM) solutions, Intrusion Detection and Prevention Systems (IDSs/IPSs), etc., are crucial to produce threat intelligence for a specific organization. Indeed, according to ENISA [PJK+14], the threat intelligence concept is associated to actionable information from the organization point of view. What can be threat intelligence for one organization is not necessary threat intelligence for another one. Bearing in mind this, **Atos research** will focus on the application of different threat intelligence criteria (such as relevance, timeliness, accuracy or variety), to analyse and correlate external cyber security data received through MISP with static and real-time data from the monitored infrastructure with the aim of qualifying it and identifying actionable information relevant for a specific organization. Additionally, the context information retrieved from the monitored infrastructure for a potential threat or attack, will allow to enrich and improve the quality of the resultant indicators of compromise.

Current intrusion detection platforms aim to analyze traffic at a central point, where the common practice is to use SSL termination to enable deep packet inspection. The downside of this approach is the lack of scalability of current solutions, in tandem with the substantial higher risk of data leakage due to the decryption of ciphered traffic (in case of a stealthy intrusion). For these reasons, **C3P** researchers will target a) modular infrastructure that provides both online and offline vulnerability detection; b) a secure distributed backend for achieving scalability, and c) integration with threat sharing platforms like MISP.

Behavior Analytics (BA) is a discipline aimed at modelling and analysing the behavior and relationships of heterogeneous entities within complex environments. BA is an important topic in different cybersecurity-oriented contexts including: fraud detection, suspicious activity detection, terrorism network discovery and malicious app detection. Behavior refers to actions and reactions of any entity (individual or system), in response to various stimuli or inputs. The recent advent of technologies for collecting and tracking behavioral data at large scale, has made it possible to devise new mathematical models that allow to analyse, understand, and predict actions. These include models for event streams, social network connections, IoT-generated events, purchasing habits and opinion formation. In this scenario, Information Sharing and Analytics for Cybersecurity (ISHA) concerns the design of AI, ML and data analytics techniques for identifying system/user anomalies and predicting/preventing security threats and adversarial attacks in large amounts of data. Within this context, **CNR** researchers devised advanced solutions for (i) Security analytics, based on behavioral profiling to detect malicious/anomalous activities; (ii) Social sensing for prediction of sensitive information diffusion flows and secure information sharing and (iii) Attack prevention/response based on ML and AI to improve reaction to incidents. In the context of CyberSecurity4Europe, CNR researchers are interested in extending the above research lines in order to develop (i) a system for identifying alternative app stores and black markets on regular and dark web; (ii) an ensemble based framework for detecting attacks in computer networks (Intrusion Detection System).

Machine learning techniques are the main tools used in intrusion detection for detecting anomalies and potential violation of pre-defined security policies. However, it is a challenging task to select an appropriate

machine learning classifier in authenticating users in practice, as the performance of a particular classifier may be fluctuant varied with the used datasets. Focused on this challenge, **DTU** will introduce an intelligent mechanism that can help select / update an appropriate machine learning classifier in a period of time. The main purpose of this mechanism is to maintain the authentication accuracy by selecting a better classifier according to different network environments.

KU Leuven will explore new enabling technologies for access control to strengthen threat intelligence platforms with more fine-grained authorization levels to mitigate to put the control back in the hands of security event producers and protecting against honest but curious platform providers. The focus will be on enhancing the security of the APIs offered by TIPs to automate data exchange across different instances and platforms, and maintaining the confidentiality of threat event information both for threat events at rest (i.e. persistence) and in transit (i.e network). This research will be carried out on top of state-of-practice open source solutions. In addition, KU Leuven will investigate the implications of (federated) machine learning in an adversarial setting.

POLITO will focus its efforts on detecting various types of new cyber-attacks plaguing the modern IT infrastructures. In particular, POLITO will focus on various kind of web attacks (e.g. SQL injections and XSS attacks) since, apart web sites, REST services and web APIS are becoming more and more common. In addition, POLITO will also pay attention on detecting various kind of malware applications such as worms and ransomware menaces, which in recent years have been threatening not only private and company users, but also government offices.

UMA will contribute with the analysis and definition of solutions in the areas of digital forensics and malware analysis. In particular, part of the effort will be used to analyse data normalization problems in the context of a digital investigation where multiple sources for digital evidence are possible. This is a huge problem considering the Internet of Things (IoT) forensics scenarios. New solutions must be proposed in order to identify relevant data and then to integrate this knowledge with the results obtained from different tools. In addition, threat intelligence services plays a decisive role, because they can help to nurture all the information deduced and vice-versa; the solutions proposed here can help to threat intelligence platforms to improve their results.

University of Murcia will explore new mechanisms for a secure, privacy-preserving and reliable Cyber Threat intelligence data sharing among CSIRTs, CERT and related entities. To this aim, UMU will devise implement, integrate and validate novel trust models specifically tailored to consider external entity's reputation and security dimensions, as well as other indicators of the shared information in order to determine the trustworthiness of the exchanged data. The research will be performed leveraging current state of the art platforms for data sharing such as MISP.

UNITN/FBK will focus its research activities on the detection and mitigation of cyber-security threats in "softwarized" networks, where the security of residential and business users can be provided by means of sets of software-based network functions running in servers or commodity hardware. In this regard, UNITN/FBK will study and implement Deep Learning (DL) methods for the detection of network threats such as DDoS attacks, brute force attacks, port scans, SQL injections, etc., and will combine them with Linux kernel-based traffic filtering mechanisms. The expected outcome is a full-fledged intrusion prevention system that can identify and block malicious network traffic with minimal consumption of the

server's computing resources. The research work will be carried out by leveraging recent DL frameworks (e.g., Keras and TensorFlow) and implemented on top of the UNITN/FBK's Edge Network Intrusion Detection System (ENIDS).

Cyber risk estimation of a network relies heavily on data collected from threat intelligence sources to determine the probability of being the target of an attack and the related likelihood of compromise. However, for advanced threat actors, the current practices of risk assessment (e.g. risk matrices) are insufficient. **UNITN** will focus on the analysis of data about Advanced Persistent Threats (TTPs, malware, and tools employed) obtained from different sharing platforms to determine formal models that describe APTs behaviour. In this regard, UNITN will study and implement methods that can employ these models to estimate the risk of compromise of a given network.

3.2 Mapping of partners' work on research topics

	ATOS	C3P	CNR	DTU	KUL	POLITO	UMA	UMU	UNITN / FBK
Mechanisms to share digital evidence					✓		✓	✓	
Threat intelligence information services	✓	✓		✓	✓		✓	✓	
Interoperability in privacy, requirements and regulation				✓	✓	✓			
Threat detection and security analytics	✓	✓	✓	✓	✓	✓	✓		✓
Security intelligence in defensive systems	✓	✓	✓	✓	✓				✓

Table 1: Mapping of project partners and research topics

From the above table, one can observe that there is ample expertise in the area of threat detection, security analytics and intelligence. Additionally, several project partners in T3.4 have knowledge on how to make these capabilities accessible as services to other parties or building blocks. Other partners have expertise in cryptography and privacy-aware machine learning techniques to apply them as a privacy enhancing technology (PET) in order to maintain the confidentiality of sensitive threat information. However, the implication in terms of interoperability at the level of regulation (e.g. mandatory notification) is an area that needs further research and exploration.

4 Application cases and use case demonstrators

This document reviews the application cases document in deliverable D4.1 [D41 2019], and highlights the impact on Task 3.4 in terms of relevant requirements and identified research challenges in the area of security intelligence.

4.1 Open Banking

The Open Banking ecosystem emerged from the Payment Services Directive 2 (PSD2) that enables bank customers, including both consumers and businesses to use third parties to manage their finances. As a result, banks are obliged to expose access to customer accounts through open APIs. This shift will impose new security requirements, as open APIs are becoming critical business components in a multi-stakeholder and multi-organization ecosystems.

At the same time, the banking sector is witnessing how threats are becoming increasingly professional and repeatable, and carried out by adversary with high skills and large resources. The need to settle transactions in real time limits the ability of banking players to efficiently react in the event of proven fraud.

As a result, there is a need for novel solutions to detect fraudulent consumption of APIs in an ecosystem where banks no longer control the business applications. More specifically, from a security intelligence point of view, there is a need for a strong ecosystem of community exchange platforms for critical information:

- Anonymization and desensitization of critical data
- Community datasets that can better train AI systems to detect fraud and threat, exploiting exchanges
- AI to profile API consumers to identity and fight against fraudulent consumptions

4.2 Supply chain security assurance

The supply chain security assurance vertical is mainly driven by the Industry 4.0 revolution where a digital transformation is taking place to improve and optimize many manufacturing processes by means of a convergence of operational technologies (OT) and information technologies (IT). Internet of Things (IoT), Cyber Physical Systems (CPS), Artificial Intelligence (AI), Big Data analytics, and cloud and edge computing are becoming part of the value chain of manufacturing.

IT solutions need to adapt to OT capabilities a.o. to add functional capacities to audit and establish accountability measures in distributed manufacturing systems to address the growing amount of threats that are the result of extending businesses through the internet.

From a security intelligence point of view, there is a need for:

- A dynamic risk assessment to select suppliers based on a systematic security evaluation

- Reliable and dynamic event management mechanisms, prevention and detection, possibly with specialized Security Information and Event Management (SIEM) solutions
- Big data, ML and AI techniques for pattern extraction and identification of unforeseen events, anomalous states, or abnormal behaviors

4.3 Privacy-preserving identity management

This demonstration case deals identity management (IdM) and the ecosystem of identity providers (IdP) and service providers (SP). The objective is to realize an identity infrastructure that is user friendly, scalable and privacy-preserving.

There is a clear need for privacy enhancing technologies (PET), but they are more situated in the area of cryptography and decentralized architectures rather than in the area of security intelligence. As such, privacy issues within the frame of this demonstration case will be alleviated with complementary enabling technologies of WP3, with are considered out of scope for the research roadmap on security intelligence.

4.4 Incident reporting

The vertical domain of incident reporting deals with creating, adopting and promoting a collaborative approach to improve, in particular, the cyber-resilience of the actors within a given sector. This implies the secure sharing of reliable threat information to detect and prevent attacks, and improving incident response.

In order to create an increased preparation and awareness in the area of cybersecurity, several challenges must be addressed. First of all, there is a lack of harmonization of procedures at the EU and national levels in terms of timeline, the data, and means of communication. Given the plethora of authorities to whom incidents must be reported, a second challenge is to facilitate the collection of threat intelligence information and data about incidents and leaks must be facilitated with a one-stop-shop approach. A third challenge is to develop models that encourage organizations and entities to share relevant information about their security incidents.

From the point of view of this vertical, there is a need for:

- Increasing the trustworthiness of the threat intelligence platforms and the actionability of the data shared to promote collaboration and voluntary info-sharing
- Open source platforms for incident reporting and impact assessment (including GDPR incidents)
- Definition of a common incident taxonomy that incorporates all regulatory requirements

4.5 Maritime transport

This vertical is characterized by collaborative and complicated processes that heavily rely on critical information infrastructures for domestic and international transportation with communications and information technology solutions for warehouse management, order and inventory control, materials handling and import/export facilitation, etc.

Key challenges in the maritime sector include a.o. the lack of information sharing, a concern that is also raised in other vertical domains. Relevant stakeholders, such port authorities, governments, supply chain providers, public authorities as well as private undertakings are reluctant to share information. The main reasons for this concern are the fear of compromising sensitive business information, their reputation, or the risk of breaching data protection rules.

From a security intelligence point of view, there is a need for:

- Solutions to detect and analyse anomalous activities and attacks in real-time
- Big data, ML and AI techniques to extract patterns in data and identify abnormal behaviors

4.6 Medical data exchange

The field of medical data exchange deals with the processing and handling of sensitive personal information by healthcare providers so that adequate patient care can be offered. The challenges in this vertical domain focus a.o. on consent and the enforcing of patient rights in compliance with the GDPR, inadequate technical measures to handle large amounts of data and the exchange of data and interoperability between cooperating companies.

The technologies sought in this domain focus on end-to-end encryption for data at rest and in transit, multi-factor authentication and access control. However, from a security intelligence point of view, there is also a need to

- Implement logging and IDS solutions that address the patient safety and privacy trade-off, limiting access to data following the least privilege principle while being able to automatically analyse logs.

4.7 Smart cities

The vertical domain of smart cities deal with the proliferation of sensors and actuators to help automate a variety of processes in the city and in the smart home. These sensors monitor significant parts of everyday life, and are a valuable target for attackers.

Key concerns are the inherent trust in smart city stakeholders and the holders of data, and the reuse of data for public services in a GDPR compliant manner. This requirements a.o. infrastructure solutions to obtain consent, to federate trust, and to secure access to data while mitigating a.o. tampering with data. From a security intelligence point of view, no additional requirements or technologies sought were identified.

4.8 Common requirements and challenges

With respect to the activities carried in the frame of T3.4, the following common requirements and challenges were identified across the different vertical domains:

- Data exchange and information sharing
- AI or Big Data to extract and identify abnormal behavior
- Unlinkability of transactions and threat intelligence challenges due to encryption and privacy enhancing technologies

5 Catalogue of enabling technologies

This section builds upon the list of enabling technologies defined in deliverable D3.1 [D31 2019], and provides a more detailed description to create a better understanding of what these assets do, how they operate, how they may complement one another, and how they may be used to collectively to strengthen one another.

5.1 Partner-specific enabling technology assets

The technology assets are listed per partner below. For a more detailed description of the labels used to describe the assets below, we refer to the Common Framework Handbook document in deliverable D3.1.

5.1.1 TIE: Threat Intelligence intEgrator (ATOS)

Name: Threat Intelligence Integrator
Partner: ATOS
Capability: Cyber Threat Intelligence
Category (L2): Security Operations Center (SOC)
Category (L3): Cyber Threat Intelligence
Type: Software: Component
Description: The Threat Intelligence intEgrator is able to correlate static and real-time information, associated to the monitored infrastructure, with cybersecurity related data coming from external OSINT sources (e.g., Indicators of Compromise), through a heuristic
Development Phase Status: Implementation

The **Threat Intelligence intEgrator (TIE)** asset, which corresponds to the Context aware intelligence sharing module of the Enriched Threat Intelligence Platform described in [Faiella 2019], will allow to integrate and correlate cyber security related information shared from external OSINT data sources or threat intelligence sharing platforms (e.g. Indicators of Compromise), with static or real-time information related to the end user infrastructure. Through a heuristic analysis process, the incoming IoCs will be enriched with an indicator of their relevance, accuracy and actionability in the end-user infrastructure. Moreover, taking advantage that the usage of the open source Malware Information Sharing Platform (MISP), the resultant IoCs with the additional information provided by this asset can be easily shared in an automated way with external trusted entities [Faiella 2019]. Figure 4 shows the Threat Intelligence intEgrator architecture with its two main elements: (i) a MISP instance and (ii) the Heuristic Module.

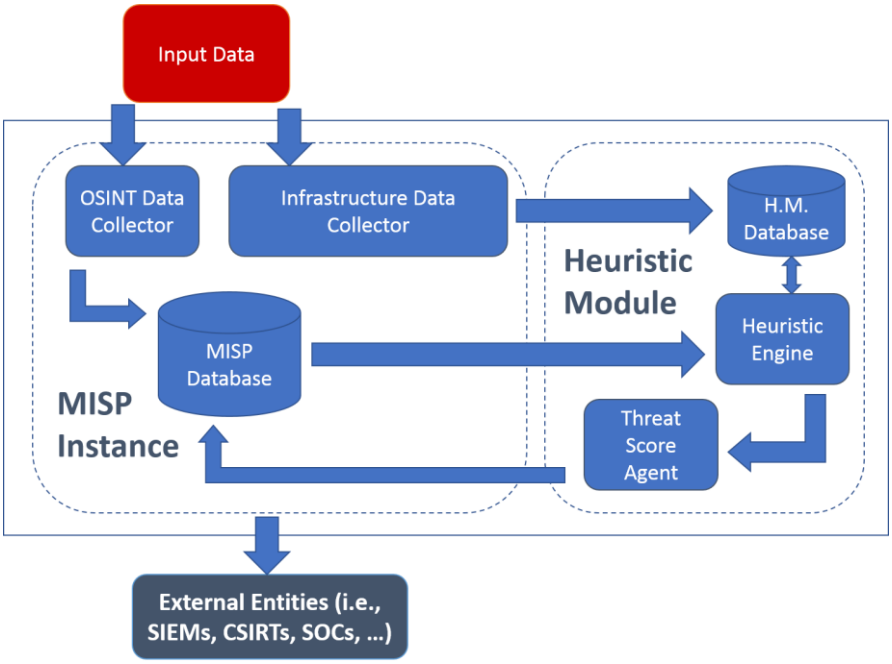

Figure 4: Threat Intelligence intEgrator Architecture [Faiella 2019]

On the one hand, through the MISP instance the asset gathers and processes in real-time external OSINT data as well as internal IoCs, security events or alerts generated by the own end-user monitoring infrastructure and static information about the end-user deployment (e.g. inventories with deployed sensors or IP addressing). On the other hand, and using the expansion modules that are available with the MISP instance if it is required, it generates enriched IoCs that can be shared to external entities. For example, they could be exported to an ArcSight SIEM using the specific MISP export module for CEF format (supported by ArchSight) [DiSIEM 2018].

The Heuristic Module, as its name indicates, it is the responsible for performing the heuristic context-aware assessment of the incoming OSINT data. The heuristics engine has configured a set of features or attributes that will be evaluated against a predefined criteria in order to determine a weight to each of them. Every feature presents in the incoming information received from MISP is evaluated against those predefined criteria and it is assigned an individual score. The aggregation of all the individual scores results into the final Threat Score (see Figure 4) that will enrich the outgoing IoCs (enriched Indicator of Compromise or eIoC). Consequently, the resultant threat score represents what is the quality of the incoming information in the sense of real intelligence useful for the end-user infrastructure. Threat scores close to zero are not reliable and should have a low priority treatment [DiSIEM 2018]. [Gonzalez-Granadillo 2019] presents an example of a case study where it is received an IoC about a specific vulnerability about a critical remote code execution that it is correlated using the heuristics engine with an inventory of the infrastructure's network and the applications already installed there.

The resultant IoCs, enriched with the information about the heuristic assessment performed, are stored in the MISP database and sent back to the MISP instance for sharing. When the external entity is also a MISP instance, the sharing process is done through the synchronization of MISP instances. In other cases, it can

be used the REST APIs available. In this way, these eIoCs can help to SOC analysts and incident response teams to know the priority and relevance of the incoming information from external OSINT data source and provide them support to the decision-making process. Additionally, they can be also used with other tools used for threat detection in order to complement the information generated by these tools and reduce the number of false positive and false negative [DiSIEM 2018].

5.1.2 Briareos (C3P)

Name: Briareos
Partner: C3P
Capability: Detect and Respond
Category (L2): Security Continuous Monitoring
Category (L3): SIEM / Event Correlation Solutions
Type: Software: Component
Description: Modular Framework for Elastic Intrusion Detection and Prevention
Development Phase Status: Design/Implementation

Briareos is a modular framework designed for improving network security by detecting and preventing intrusions. It achieves this by extending rule-base *Network Intrusion Detection Systems* (NIDSs), which detect attack vectors with a known signature (i.e., already reflected in existing rules), with the capability of inferring new rules for unknown attack vectors (not covered by existing rules) by inspecting network traffic (both inbound and outbound) and correlating it with operating system behavior. Its modular architecture allows Briareos to provide users with the possibility of building new modules and creating new processing pipelines for handling intricate intrusion schemes, not possible by traditional rule-based systems. In essence, Briareos allows networks and systems to be increasingly secure over time by detecting and classifying unknown attacks and preventing system from future attack occurrences. Focusing on performance, Briareos offers adjustable security levels that provide system administrators a tradeoff between performance and the level of security intended for a host or a set of hosts.

Architecture

Briareos architecture, as depicted in Figure 5, is comprised of different modules, including: the *Briareos Host Component*; the *Briareos Manager Server*; and, an optional *Distributed Offloading* component.

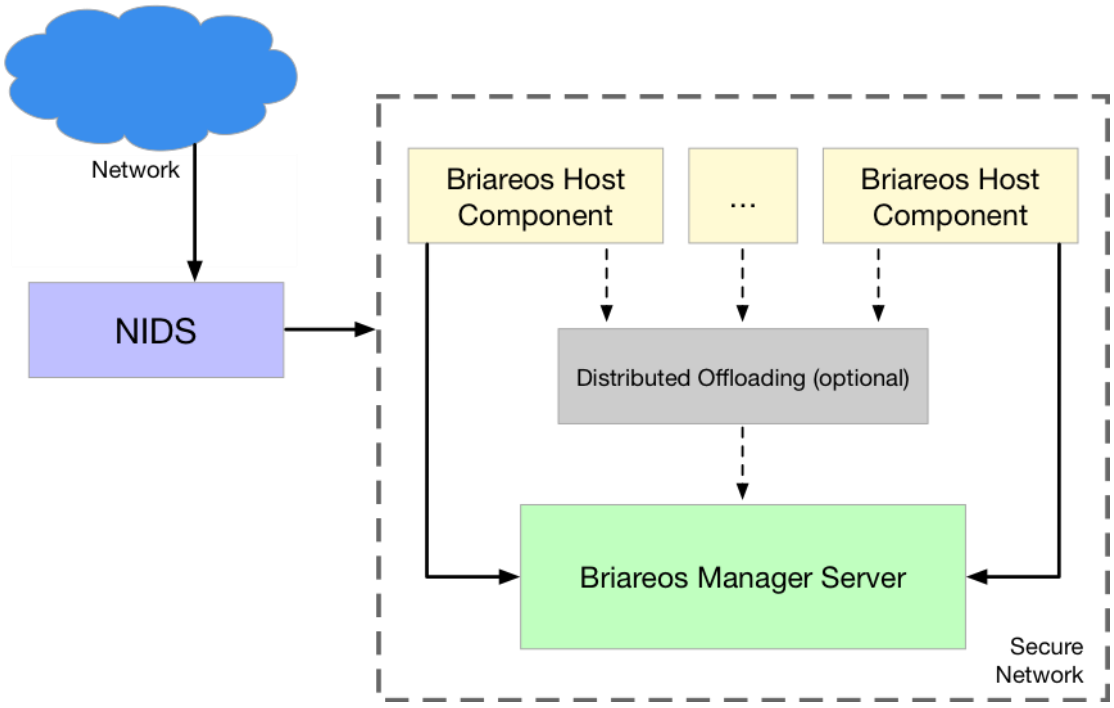

Figure 5: Briareos architecture

Briareos Host Component

Each host on the network is protected by the Briareos Host Component (BHC).

This component is responsible for:

- intercepting and inspecting ingoing network traffic;
- intercepting and inspecting outgoing network traffic; and,
- processing traffic (both inbound and outbound) in order to detect intrusions.

Processing engine

Traffic processing is accomplished by the BHC's Processing Engine (PE). During startup, PEs load configured pipelines (and respective modules), associate pipelines with network ports and traffic type (inbound and/or outbound). Pipelines are responsible for analyzing intercepted traffic, dropping invalid packets, i.e., packets that do not comply with the pipeline's rules. Multiple individual pipelines can be composed into larger pipeline groups, where only valid packets are passed from one individual pipeline to another.

Briareos offers different processing schemes: *inline*, *parallel*, or *hybrid*. The inline processing scheme uses a *streaming processing approach* where inbound traffic is processed before being delivered to the respective service, and outbound traffic is processed before being sent to the network. This offers the strictest security level by dropping invalid packet before reaching the respective service, and preventing sensitive information from being leaked to the network. However, this increases service latency due to processing delays.

The parallel processing scheme delivers inbound traffic in parallel to the service and the BHC, and outbound traffic in parallel to the BHC and the network. This improves latency, since traffic processing is performed concurrently with the service, however it allows small intrusion windows to become exploitable, since unknown attacks can only be detected and prevented possibly after occurring, since these execute concurrently.

The hybrid approach mixes both schemes, allowing inbound traffic to be processed inline and outgoing traffic to be processed in parallel, or vice-versa.

To improve Briareos performance and scale to large networks, Briareos uses a centralized knowledge broker, called *Briareos Manager Server* (BMS), that is responsible for maintaining an aggregated knowledge base of all attack signatures, BHCs share their knowledge base with the (BMS).

Briareos Manager Server

The Briareos Manager Server is responsible for gathering and merging the different BHCs knowledge base, and disseminating the resulting knowledge base to BHCs and/or NIDS.

If a given host is the target of an unknown attack and detects or prevents it, then all the other hosts will be automatically protected. For example, every BHC can create new signature-based rules and share them with the other hosts in the network. The BMS is the component that distributes intelligence through the BHCs and the NIDS. The BMS is responsible for collecting new rules from hosts and also publishing local rules to every BHC that subscribes the feed. On the other hand, if a given rule is a global rule, it will be propagated to the NIDS instead. It is also important to mention that the BDS can also create new rules and push them to the BMS.

Distributed Offloading

Complex traffic processing schemes, such as data-mining, can be processed in parallel using the Briareos Distributed System (BDS). This allows the workload to be shared among multiple workers thus reducing resource consumption of the hosts.

5.1.3 UASD: Unauthorized App Store Discovery (CNR)

Name: UASD - Unauthorized App Store Discovery
Partner: CNR
Capability: Detect
Category (L2): Detection Processes
Category (L3): Underground/Darkweb investigation
Type: Software: Component
Description: Allows to identify unauthorized mobile app stores (black market) in regular and dark web.
Development Phase Status: Testing

UASD - Unauthorized App Store Discovery. Nowadays, implementing brand protection strategies has become a necessity for enterprises delivering services through dedicated apps. Increasingly, malicious developers spread unauthorized (fake, malicious, obsolete or deprecated) mobile apps through alternative distribution channels and marketplaces. Most of the current approaches assume prior knowledge about the

marketplaces to be monitored. The UASD asset is a framework for the early detection of these alternative markets advertised through social media such as Twitter or Facebook or hosted in the Dark Web.

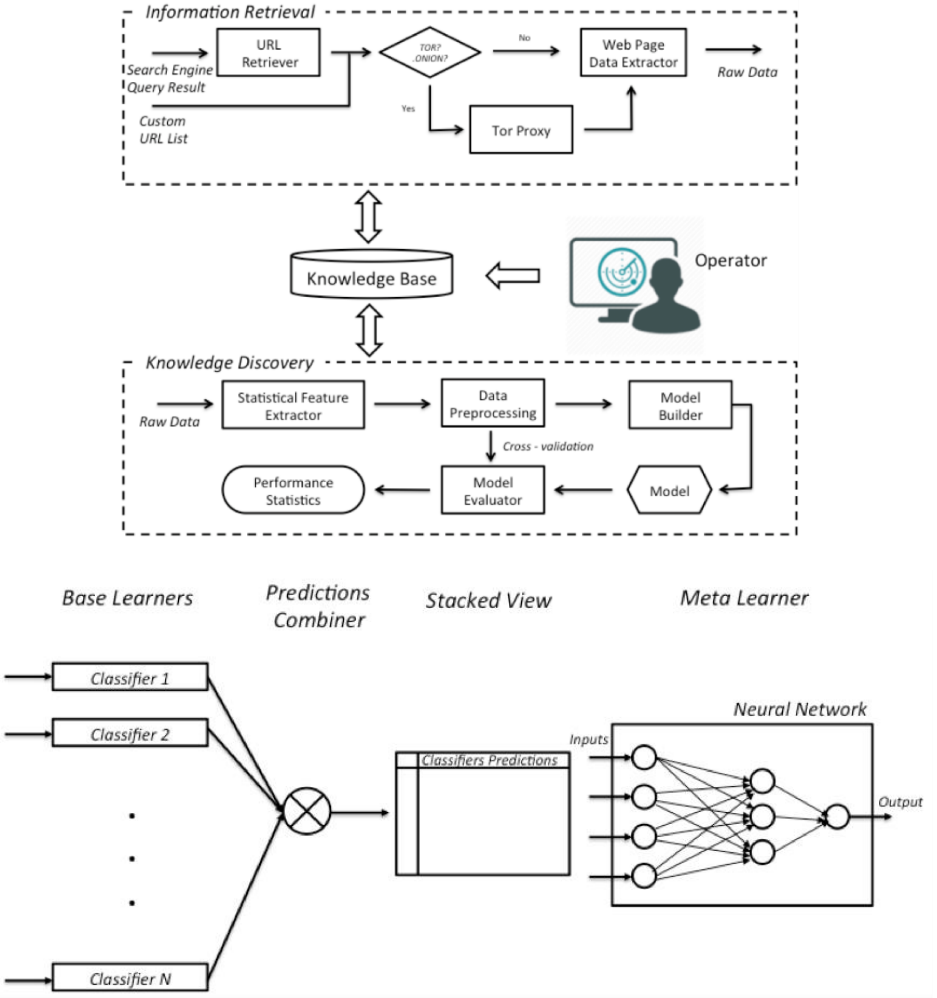

Figure 6: (a) UASD Framework Architecture (b) UASD Classification/Prediction model

Specifically, it is meant to combine a data modeling approach and an ensemble learning technique, allowing to recommend web pages that are likely to represent alternative marketplaces or black market. UASD allows to analyze web pages extracted from the Web and, by exploiting a classification model, to distinguish between real app stores and similar pages (i.e. blogs, forums, etc.) which can be erroneously returned by a common search engine. The framework (see Figure 6) is composed by three main macro components: Information Retrieval, Knowledge Discovery and Interaction with the operator. Human experts devise a set of web queries by exploiting Advanced Google Search Operators or specify some URLs in order to identify possible alternative mobile markets in regular Web or in the TOR network (Dark Web) by exploiting the Information Retrieval component. Raw data are stored in a shared Knowledge Base. Knowledge Discovery

component allows to learn a classification/prediction model gathered by the information retrieval module. We can devise three components in it: (*i*) Data Transformation (*ii*) Prediction Model and (*iii*) Evaluation. Different types of classifiers are combined according to a stacking schema to build an ensemble model. In detail, the predictions of these base learners are combined in a stacked view (where each column contains the prediction provided by a single classifier) that feeds an Artificial Neural Network (ANN) as a meta-learner. Finally, the overall process is monitored by human experts, as described which represents the steady and operational states of the system.

5.1.4 EBIDS: Ensemble Based Intrusion Detection System (CNR)

Name: EBIDS - Ensemble Based Intrusion Detection System
Partner: CNR
Capability: Detect
Category (L2): Detection Processes
Category(L3): Intrusion Detection
Type: Software: Algorithm
Description: The tool is an ensemble-based approach used to identify anomalous/suspicious activities in networks and computer systems
Development Phase Status: Implementation

EBIDS - Ensemble Based Intrusion Detection System. Modern intrusion detection systems must handle many complicated issues in real-time, as they have to cope with a real data stream; indeed, for the task of classification, typically the classes are unbalanced and, in addition, they have to cope with distributed attacks and they have to quickly react to changes in the data. Data mining techniques and, in particular, ensemble of classifiers allow to combine different classifiers that together provide complementary information and can be built in an incremental way. We will define an intrusion detection framework (see Figure 7) where the detector module is based on a meta-ensemble, which is used to cope with the problem of detecting intrusions, in which typically the number of attacks is minor than the number of normal connections. In particular, we will explore the usage of ensembles specialized to detect particular types of attack or normal connections and deep learning based methods to ensure accurate predictions. The current implementation of the detection framework relies on the induction and exploitation of multiple base classifiers, which are expected to provide different, hopefully complementary, models for a given number of targeted attack types. These base classifiers take the form of Deep Neural Networks (DNNs) sharing all the same architecture, but trained against different samples of the given training data. These base classifiers are reused as parts of an ensemble model adhering to a deeper and more complex DNN architecture, where a number of layers are devoted to combine different kinds of signals produced by the base classifiers. A two-phase learning procedure is exploited for training the ensemble model: first, different instances of the base DNN architecture are trained separately, over different data samples, in order to build the initial versions of the base classifiers. Then, the overall ensemble model is trained, over a small subset of examples, in order to learn how to effectively combine (and possibly fine-tune) the base classifiers discovered before.

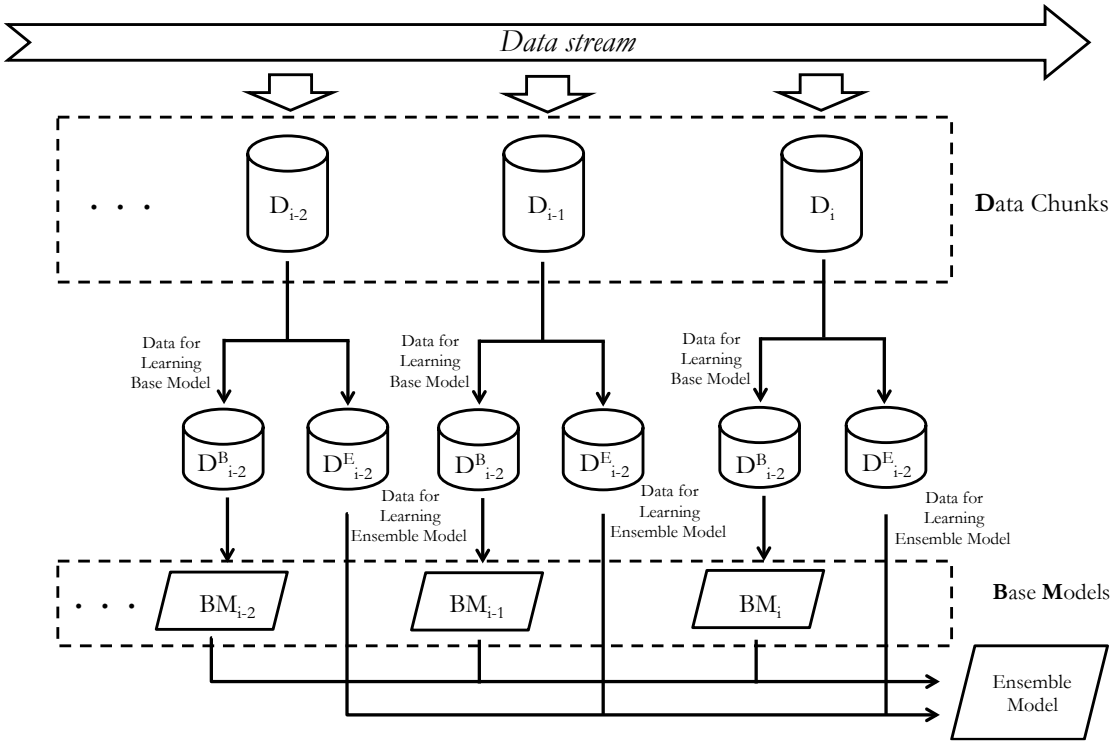

Figure 7: EBIDS Incremental Learning Flow

5.1.5 IntelFrame: Intelligent Machine Learning-based Intrusion Detection (DTU)

Name: IntelFrame - A Framework for Intelligent Machine Learning-based Intrusion Detection
Partner: DTU
Capability: Detect
Category (L2): Detection Processes
Category (L3): Intrusion Detection
Type: Software: Component
Description: This framework allows each IDS node to select an appropriate machine learning algorithm from a pool in a periodic manner, with the purpose of maintaining the detection accuracy.
Development Phase Status: Design

In practice, the performance of a classifier would be fluctuant based on different data sources. The framework of IntelFrame aims to enhance / maintain the detection performance of an IDS, by helping choose a proper machine learning algorithm in a time period. Figure 8: An overview of IntelFrame with detailed interactions shows the framework overview and interaction details. The framework can provide several advantages:

- The algorithm can be selected intelligently in a period time.
- The algorithm with better performance will be selected.

- Algorithms can also be used to decide an anomaly via data fusion & aggregation, i.e., considering the outputs from all used algorithms.

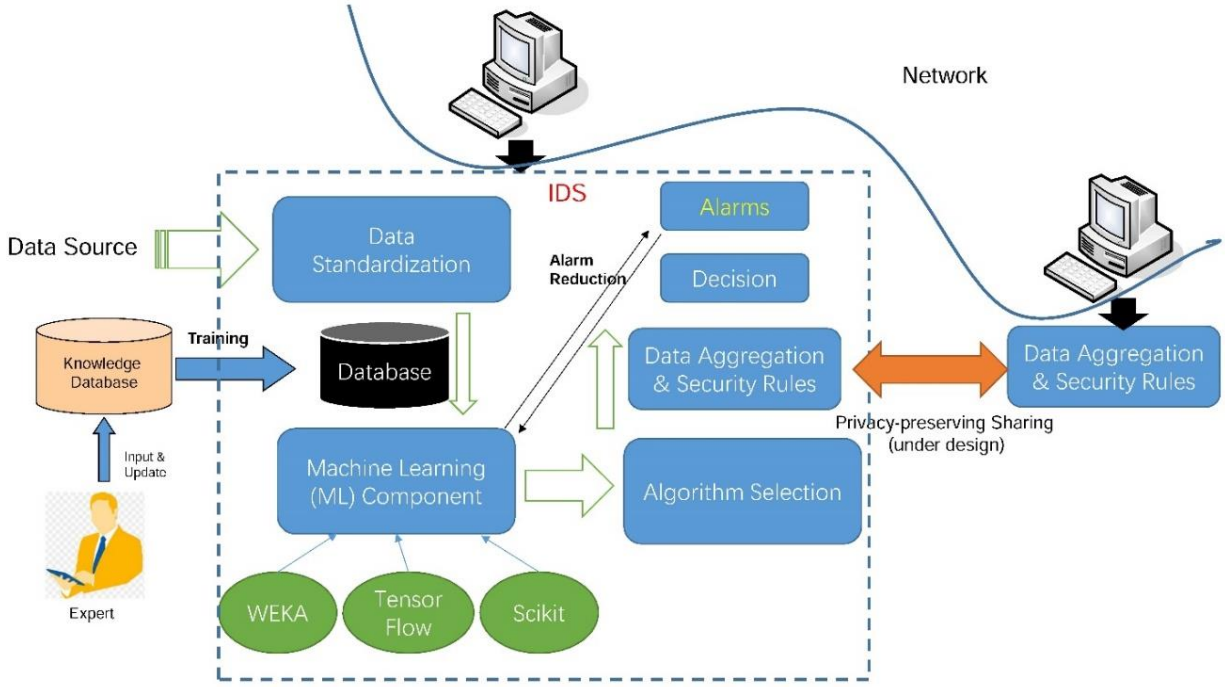

Figure 8: An overview of IntelFrame with detailed interactions

When the traffic arrives at an IDS, it will examine its payload with its own rules, or perform an anomaly detection. For the generated alarms, our framework can intelligently select a suitable classifier for false alarm reduction. The merit is that the classifier could be adaptive and distinct for different IDS nodes.

For selecting the classifier, we can input traditional supervised learning and ensemble learning algorithms. Expert knowledge can also be used to help enhance the labelled database for both traffic anomaly detection and false alarm reduction. IntelFrame can also adopt a security rule sharing component, which can use crypto methods to provide privacy-preserving sharing between IDSs, e.g., Rabin fingerprint algorithm.

Figure 9 depicts the detection workflow under IntelFrame, including data standardization, machine learning classifier training and selection, data aggregation, output decision.

- **Data standardization.** To ensure the data can fit various machine learning algorithms, the first step is to standardize the raw data by extracting pre-defined features and reducing noisy data. According to different machine learning tools, such as WEKA (https://www.cs.waikato.ac.nz/ml/weka/), Tensor Flow (https://www.tensorflow.org/) and Scikit (https://scikit-learn.org/stable/), there is a need to adopt particular data format.
- **Machine learning classifier training.** The standardized data can be stored in a database, and then can be used to train the classifiers in a pool. For different machine learning tools, they may contain distinct classifiers. Security manager can decide which classifier(s) can be maintained in the pool.

- **Machine learning classifier selection.** By training and building all classifier models, the key step is to select a classifier that provides better performance in an intelligent way. Security manager can configure the selection settings. 1) Security manager can choose required metrics to help choose a better classifier, in terms of accuracy, time consumption and cost. 2) Security manager can also decide the time period in re-training and re-selecting a classifier.
- **Data aggregation.** In a distributed intrusion detection environment, various IDS nodes can exchange required data like alarms. IntelFrame can consider aggregated data and security rules to help refine and tune machine learning classifiers. It is especially useful to help choose a classifier, when several classifiers have a similar performance.
- **Output decision.** Through considering all aggregated data / security rules and the selected algorithm, a decision can be finally made to identify potential threats.

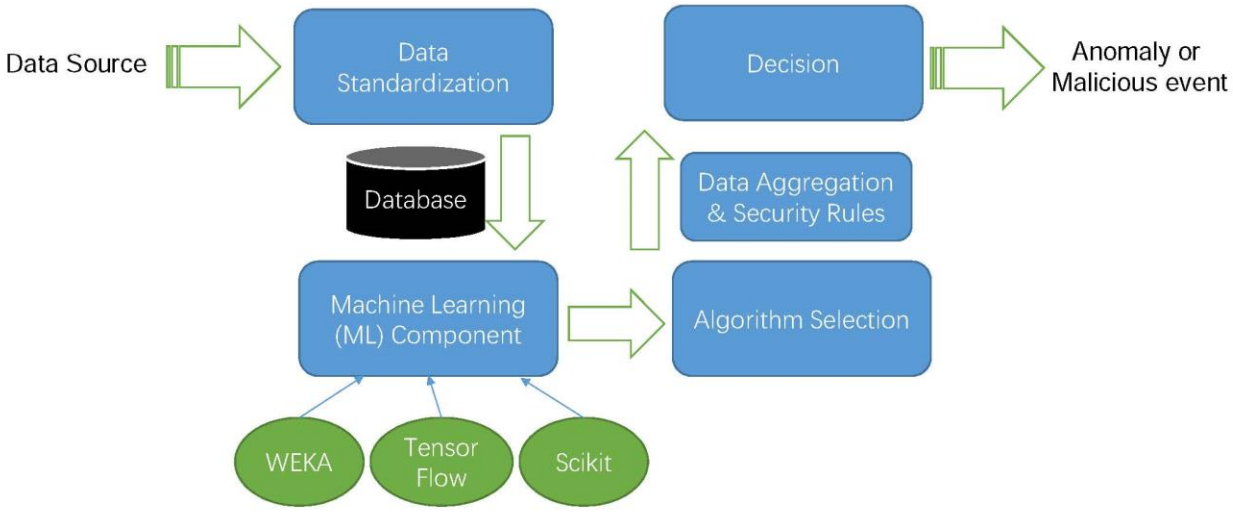

Figure 9: The detection workflow under IntelFrame

IntelFrame can provide much flexibility, while there are some challenges. i) Some additional workload is needed, as there is a need to maintain / train a pool of classifiers and to choose a particular classifier periodically. ii) Under different network environments, security manager needs to select potential classifiers into the pool. iii) There is a need to decide the timeframe to update and re-select a classifier.

5.1.6 TATIS: Trustworthy APIs for enhanced threat intelligence sharing (KUL)

Name: TATIS - Trustworthy APIs for enhanced threat intelligence sharing
Partner: KUL
Capability: Detect
Category (L2): Security Continuous Monitoring
Category (L3): SIEM/Event Correlation Solutions ; Cyber Threat Intelligence
Type: Software: Component

Description: Enhanced open source threat intelligence sharing platform to share indicators of compromise in trustworthy manner on top of the MISP platform **Development Phase Status:** Implementation

The TLP scheme is only a labeling scheme. It therefore requires a certain level of trust in the system in that all parties (TIP provider, threat event producers and consumers, etc.) are assumed to adhere to the protocol. TIPs protect a treasure trove of sensitive and confidential information, and expose data exchange, search and analytics capabilities through RESTful APIs. The security of these API endpoints becomes a critical concern. Indeed, many of such enabling technologies that are used to improve the security of enterprise systems and networks, are also being targeted. Adversaries may abuse sensitive information within the TIP to attack organizations, use the information to evade detection, or poison the data to harm communities. In their report, ENISA confirmed TIPs do not only offer opportunities, but also come with limitations. Our work mainly addresses the trust related issues identified in the report:

1. The event producer trusts the platform provider to not expose confidential data to unauthorized recipients.
2. The event producer trusts the event consumers that they handle shared information according a predefined protocol (e.g. TLP).
3. The platform provider and event consumers trust the event producer that the information shared is reliable and credible.

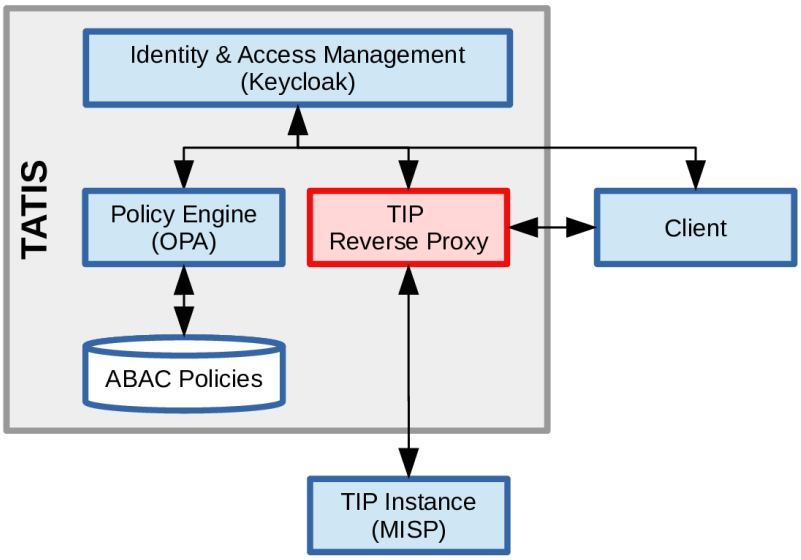

Figure 10: High-level overview of TATIS

We present TATIS, as depicted in Figure 10, a solution for fine-grained access control to protect threat intelligence APIs using User Managed Access (UMA) and Ciphertext-Policy Attribute-Based Encryption (CP-ABE). The main contributions of this work are (1) protection against honest but curious threat intelligence platform providers and data leakage through vulnerabilities in the TIP itself, and (2) more fine-grained access management by the owner or provider of threat events when offered through APIs under the control of a potentially curious platform provider.

This way, TATIS combines the strengths of UMA 2.0 (see Figure 11) and an attribute-based access control (ABAC) model implemented with externalized OPA policies. It is much more flexible and granular compared to the role-based access control (RBAC) model typically offered in TIPs, including MISP:

1. More sophisticated policies incorporate dynamic attributes, such as time, location or usage statistics for more fine grained access control decisions.
2. The owner of the data can define different privileges per API method for each user, even disabling certain methods in response to an attack.
3. When the policy denies access to a requester, the owner can still grant access, hereby bypassing the policy. This consent can be revoked by the owner.

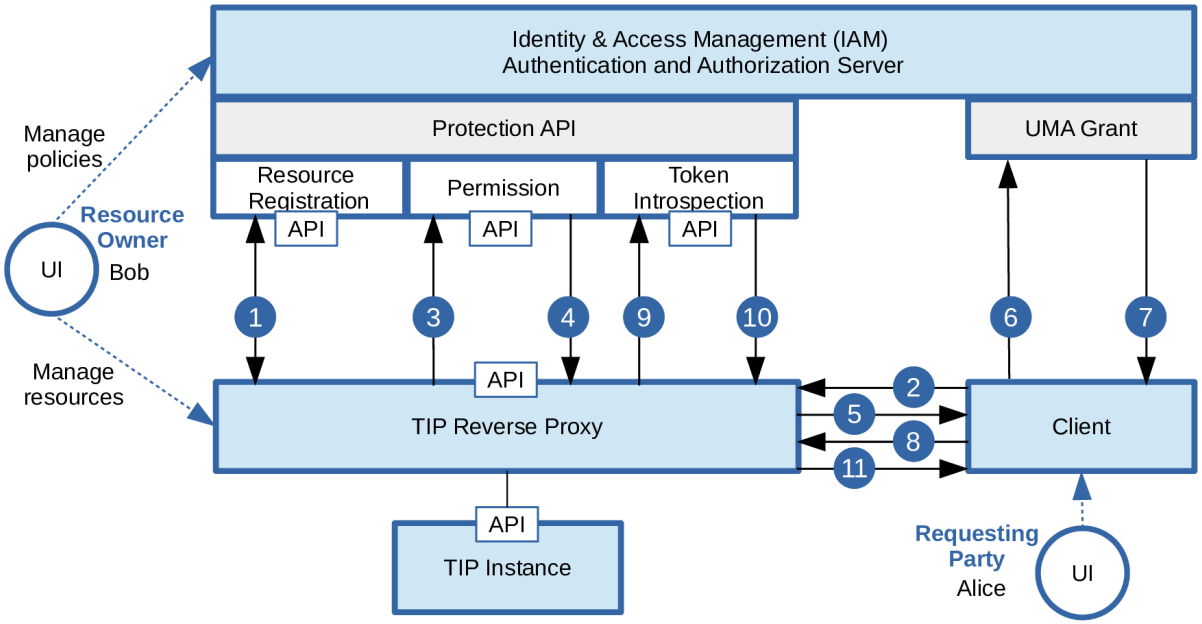

Figure 11: UMA-based access control to APIs

However, the above does not guarantee trust in the platform provider hosting the threat intelligence sharing platform. A malicious platform provider or administrator is still able to directly access the underlying database (i.e. MariaDB in the case of MISP) and gain access to all the sensitive and confidential information. To further protect the threat events stored within the database, we consider an honest-but-curious adversary model, i.e. parties that are curious and try to find out as much as possible about the sensitive threat events despite following the protocol. The TIP provider or cloud infrastructure operator can be considered as such candidate adversaries. Additionally, a vulnerability in the TIP itself may grant unauthorized subjects access to sensitive information.

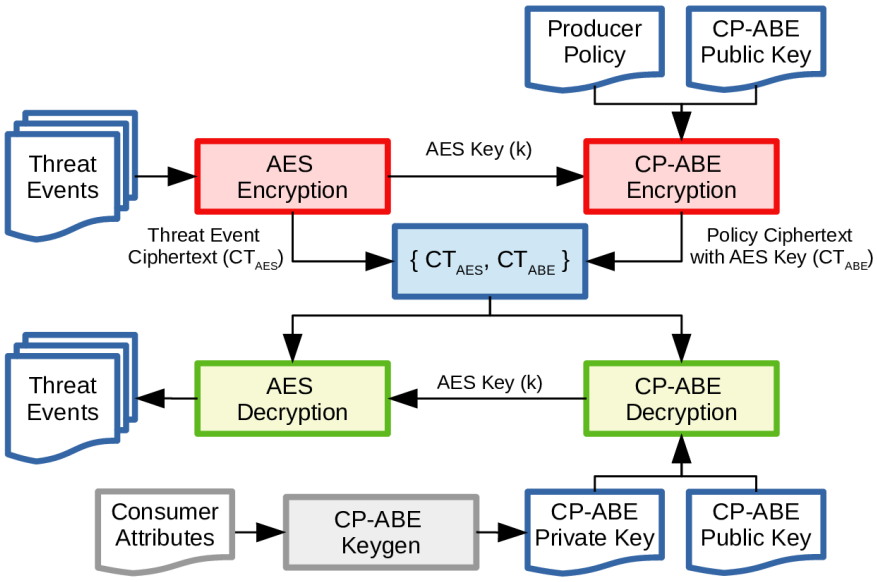

Figure 12: CP-ABE based protection of sensitive threat intelligence data

Our solution aims to encrypt events and attributes so that the confidentiality is guaranteed w.r.t. unauthorized subjects, but at the same time the event producer still wants to grant access to event consumers based on his own authorization policies. Obviously, the event producer cannot use a common encryption key for all event consumers. For scalability reasons, the event producer cannot encrypt the same information multiple times with the different public keys belonging to the event consumers. The TIP Reverse Proxy component of TATIS solves this through Ciphertext-Policy Attribute-Based Encryption (CP-ABE) [2], as illustrated in Figure 12 above. Bob encrypts the threat events with an AES symmetric key k into a ciphertext CT_{AES}, and uses a policy (i.e. a boolean access structure resembling a decision tree based on user attributes) and CP-ABE to encrypt the AES symmetric key k into a ciphertext CT_{ABE} . A user's private decryption key is linked to a set of user attributes that represent this user's permissions to decrypt. So, Alice can only decrypt the event ciphertext CT_{AES} if she can obtain the AES key k. To do so, she must decrypt CT_{ABE} with a CP-ABE private key generated from her set of user attributes. The decryption will fail if Alice's attributes do not match the access structure defined by Bob's policy.

5.1.7 NetGen (POLITO)

Name: NetGen
Partner: POLITO
Capability: Detect
Category (L2): Security Continuous Monitoring
Category (L3): Cyber Threat Intelligence
Type: Software: Component
Description: This tool generates a non-DPI analyzer that can classify any kind of network traffic
Development Phase Status: Implementation/Testing

NetGen is a tool written in Python 3 that is able to automatize the creation of network traffic analyzers using deep learning without being an expert in machine learning. Figure 13 shows a simplified version of the work-flow used by NetGen to train a neural network for traffic classification.

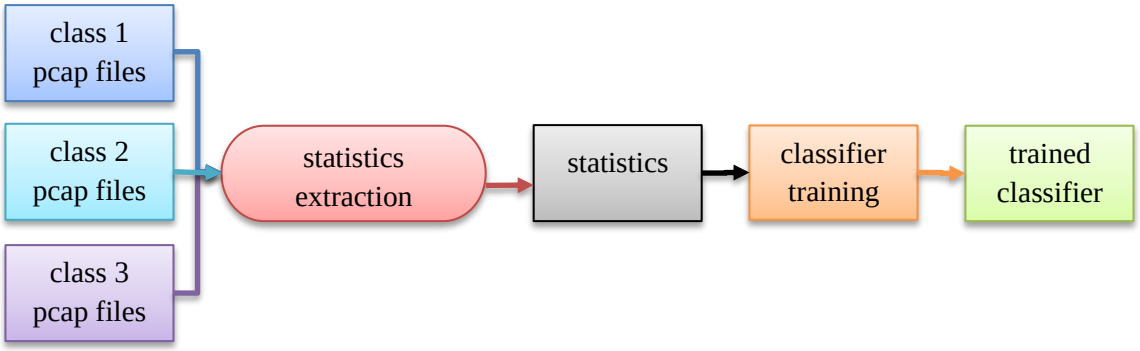

Figure 13: Network classifier training work-flow

The input of the tool is a set of pcap capture files, split for each class that needs to be identified. Once the tool is executed, it extracts a set of TCP/IP statistics for each TCP connection in the capture files using the tstat[20] network analysis tool. These statistics are then used as the input features to train a fully connected neural network, by leveraging the Pytorch[21] deep learning framework. The tool automatically performs an optimization phase to choose the best hyper-parameters for the neural network without any external intervention. In addition, it can make use of CUDA-ready GPUs to significantly speed up the training process. The final classifier is then saved to a file, ready to be used.

The built neural network can now be used to identify attacks, anomalous connections or other kind of suspicious network activities. Figure 14 depicts the traffic analysis phase of a trained classifier.

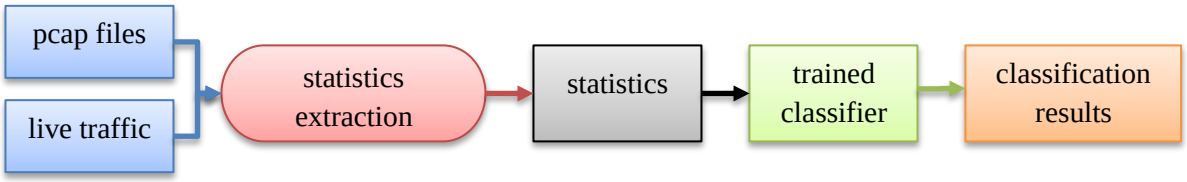

Figure 14: Network classification work-flow

A trained network analyzer can work on a pre-recorded capture file, but can also be used to classify live traffic by listening on a specific network interface. The traffic is analyzed in two sequential phases. First, its statistics are computed via tstat, then the neural network is used to produce the final classification. The results contains not only the identified class for each TCP flow, but also a confidence level (a percentage)

[20] See http://tstat.polito.it/.
[21] See https://pytorch.org/.

of the classification. The results of the classification can be printed on the standard output, saved to a file, but the developer can also directly use the network classifier as a Python module by using its internal classes.

The classification results can now be then utilized for a proper reaction, if needed, such as putting a drop/reject rule into a firewall and/or sending a warning message to an administrator.

5.1.8 JUDAS: JSON Users and Device analysis tool (UMA)

Name: JUDAS - JSON Users and Device analysis (JUDAS) tool
Partner: UMA
Capability: Detect
Category (L2): Security Continuous Monitoring
Category (L3): SIEM/Event Correlation Solutions ; Cyber Threat Intelligence
Type: Software: Component
Description: This tool collects the files to be processed, extracts relevant data and correlates them, additionally asking external services to complete the information about the objects generated (e.g. ipapi, VirusTotal, Pipl)
Development Phase Status: Design/Implementation

Digital forensic tools (DFT) can be designed to be used during the whole lifecycle of a digital investigation or to carry on specific analysis on data. Therefore, in general, DFT can be defined for a phase or set of phases of a digital investigation, depending on the scenario [Nieto 2019]. These phases can be classified, according ISO/IEC 27037:2012 and ISO/IEC 27042:2015 in eight phases, from the identification of the potential sources of digital evidence to the reporting (c.f. Figure 15). While the tools for the first four phases devised to the identification, collection, acquisition and preservation of digital evidence, are prepared to be in direct contact with the sources of potential digital evidence (e.g. hard drive, sensors, networks, etc.), the tools for the last four phases are devised to help during the analysis of digital evidence, and will operate on digital copies of the first sources. The **JSON Users and Device AnalysiS (JUDAS)** tool belongs to the second classification of tools, where the main objective is to process digital data and to extract valuable information in the context of a use case in the scope of a digital investigation.

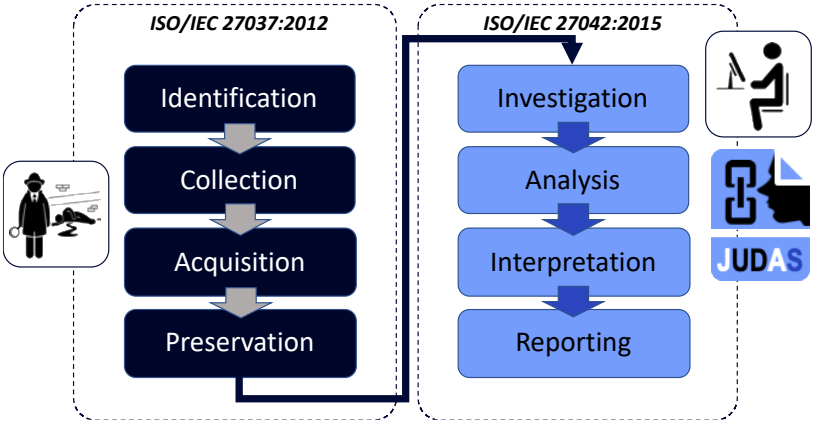

Figure 15: Phases for digital investigation in ISO/IEC 27037:2012 and ISO/IEC 27042:2015

Traditional approaches for the analysis of digital evidence focus on the analysis of specific data in isolated platforms prepared for the analysis. Well stablished tools such as EnCase and Access Data allows to conduct the digital investigation on several type of devices and also helps to build a report after the analysis. Also, the investigator can use additional tools to carry on specific analysis on certain data, such as the framework Volatility is used for memory forensics. However, is the digital investigator who has to build the context of the digital investigation, and if necessary to search for additional sources of data on the Internet. OSINT techniques are used to search for additional information about individuals or even objects.

JUDAS proposes a new methodology to create a visual representation of a digital investigation taking as a starting point the JSON files inside a folder that contains all the digital evidence for a digital investigation. Then, after generating a basic context, this context is feed with additional information acquired either i) from the analysis of additional digital evidence in the same folder or ii) from the information collected from OSINT services and threat intelligence platforms.

The decision about start from JSON files is because some scenarios in IoT-Forensics generates thousands of logs in said format (e.g. Alexa Cloud). Not only that, but also this format is chosen by multiple DFT to store the results of the operations or as summary of these (e.g. dumpit). Then, these files can help to build a general view of the context that must be fed by additional tools. The general idea is shown in Figure 16. JUDAS is modular, relying on diverse modules to analyse specific formats, and will generate a representation of a set of files. One of the main issues to integrate all the data from different tools is data normalization. This problem will also be present in the exchange of digital evidence. In this case, JSON are also used with this purpose. JUDAS defines a set of characteristics to be satisfied by the objects generated in the model, and builds the objects from different files following a set of criteria defined in the JUDAS methodology (c.f. Figure 17).

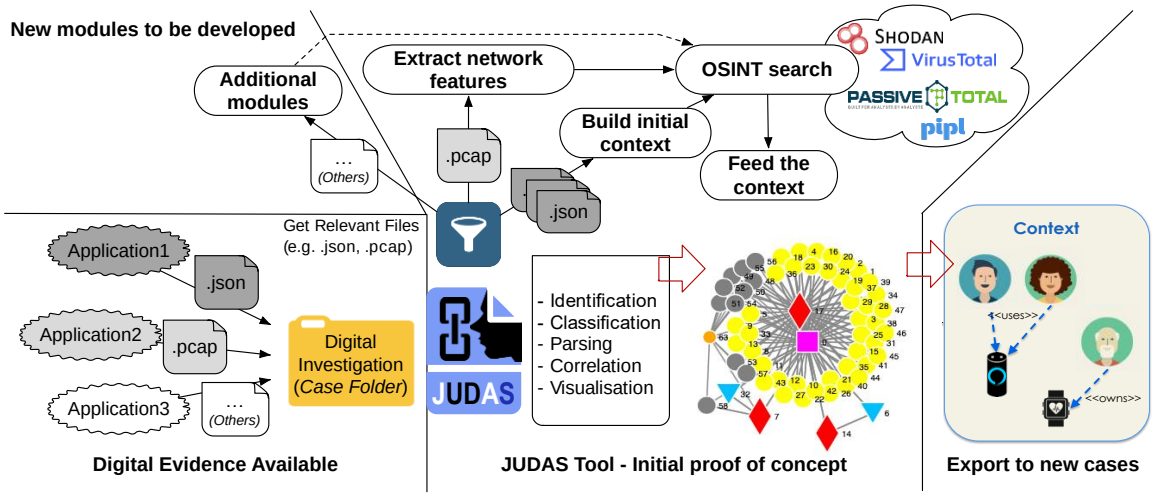

Figure 16: JUDAS – general idea and main scope

The JUDAS methodology in Figure 17 defines six main phases: identification, classification, parsing, correlation, feeding (shown as *OSINT sources & threat* intelligence) and visualization. During the identification, the content of the files is analysed in order to identify relevant keywords. The definition of keywords as identifiers helps classify the data into different objects in the second phase. In the classification

the objects of interest are defined, and then parsed to synthesise the data in unique objects. Then, during the parsing the correlation begins to match the objects with the current context that is generated while the folder is being processed. During the correlation some objects can be directly fed with OSINT and threat intelligence sources, or, instead, the user (the digital investigator) can select the objects to be processed using OSINT sources. If this option is chosen, then the classification, parsing and correlation must be performed again. It is important to highlight that during the identification and classification JUDAS is addressing the data normalization problem for the tools used in this context.

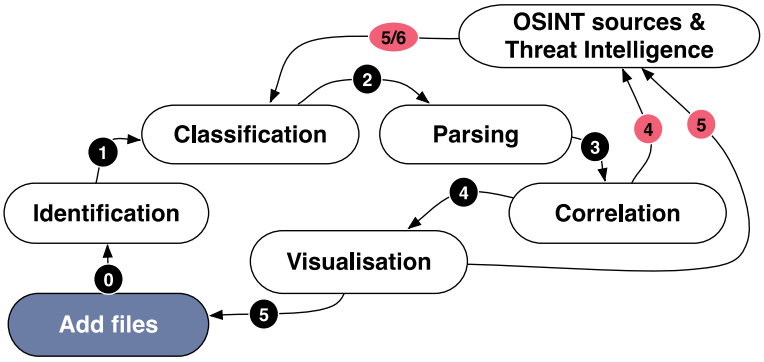

Figure 17: JUDAS methodology

A first prototype of this idea has been developed using Python (Figure 18). The main module, denote *eatingJSON* is where all the functions for classification, parsing and correlation resides, the core of JUDAS. Then, the visualization of data is performed using a GUI. This GUI also has embedded the OSINT searcher, where the keys for the API requests can be uploaded using a file. Then, additional modules are developed to process other digital evidence in the use case folder, for example .pcap files and memory dumps. In general JUDAS is prepared to continue growing in a modular faction using extensions.

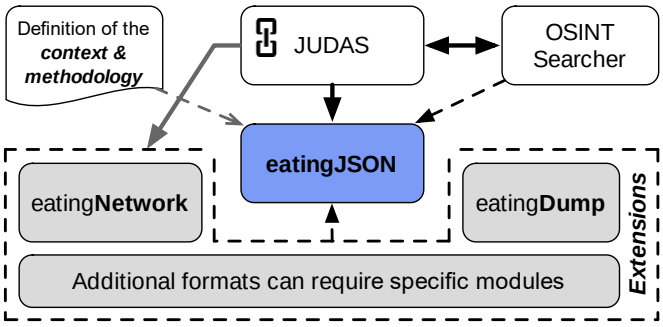

Figure 18: JUDAS development

To test the approach, a set of files from an Alexa system are selected and processed using the JUDAS tool. The initial context generated considering only JSON files is shown in Figure 19, left. The number of instances processed by the tool are shown in Figure 19Figure 19: Some results using the Alexa ecosystem, right. Initial conclusions are that there is only one recognized user in the context (one square), which is linked with three devices. However, only one of the devices is related with most of the activities, which is the Amazon Echo device, and the other two devices identified are probably personal devices. Moreover, the bar char in Figure 19 Figure 19: Some results using the Alexa ecosystemshows the number of instances

generated by JUDAS. This can be improved in the future. It is notorious that JUDAS generates 232 objects from the files processed that finally represents the same user.

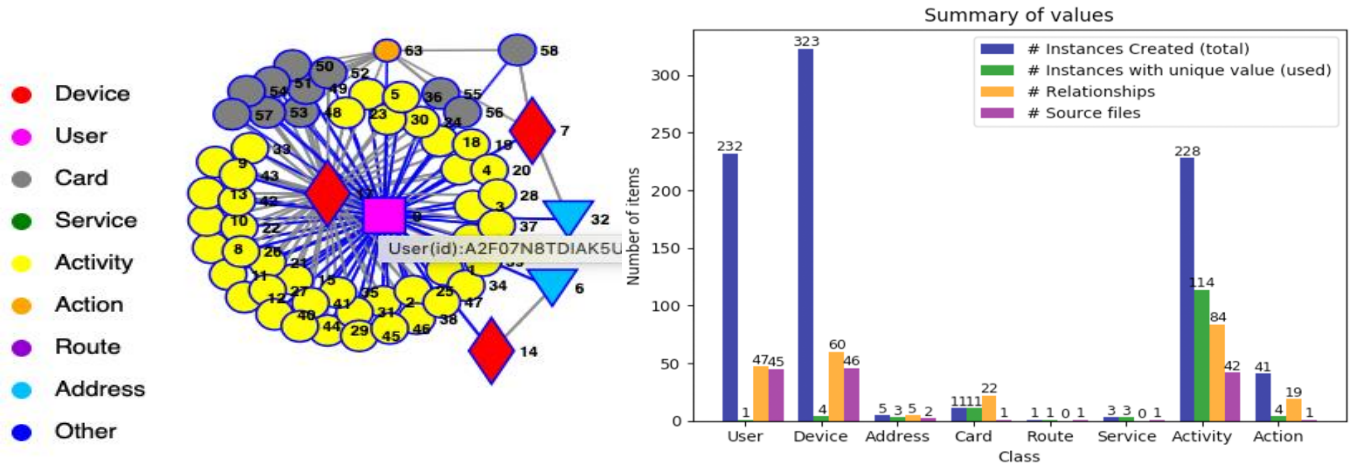

Figure 19: Some results using the Alexa ecosystem

5.1.9 HADES: Automatic analysis of malware samples (UMA)

Name: HADES - Automatic analysis of malware samples
Partner: UMA
Capability: Detect
Category (L2): Detection Process; Security Continuous Monitoring
Category (L3): Honeypots / Cybertraps; Cyber Threat Intelligence
Type: Software: Component
Description: Hades is a platform for the orchestration of sandboxes for malware execution. It can send samples to virtual machines, execute them, analyze the behaviour and create reports based on the proof generated.
Development Phase Status: Design/Implementation

HADES is an architecture designed to provide a single environment for the deployment and monitorization of honeypots. HADES uses Cuckoo Sandbox as the main orchestration component for the deployment of high interaction honeypots (c.f. Figure 20). Moreover, there are also some other low and medium interaction honeypots deployed and connected to a single point for the visualization of data. Elastic Search is used for receiving all the activity deployed in the sandboxes and Kibana presentes a configurable visualization environment of all the data gathered. Some previous tests with ELK (Elastic search, Logstash and Kibana) for this purpose were published in [Acien 2018].

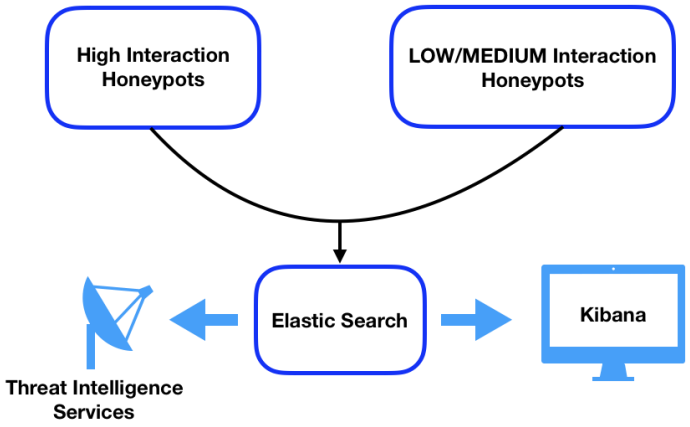

Figure 20: HADES platform

All the information gathered from these honeypots will be processed and sintetized in such a way that a final report could be integrated it into Threat Intelligence Services, like CIRCL using the MISP API. Some of these results are shown in Figure 21.

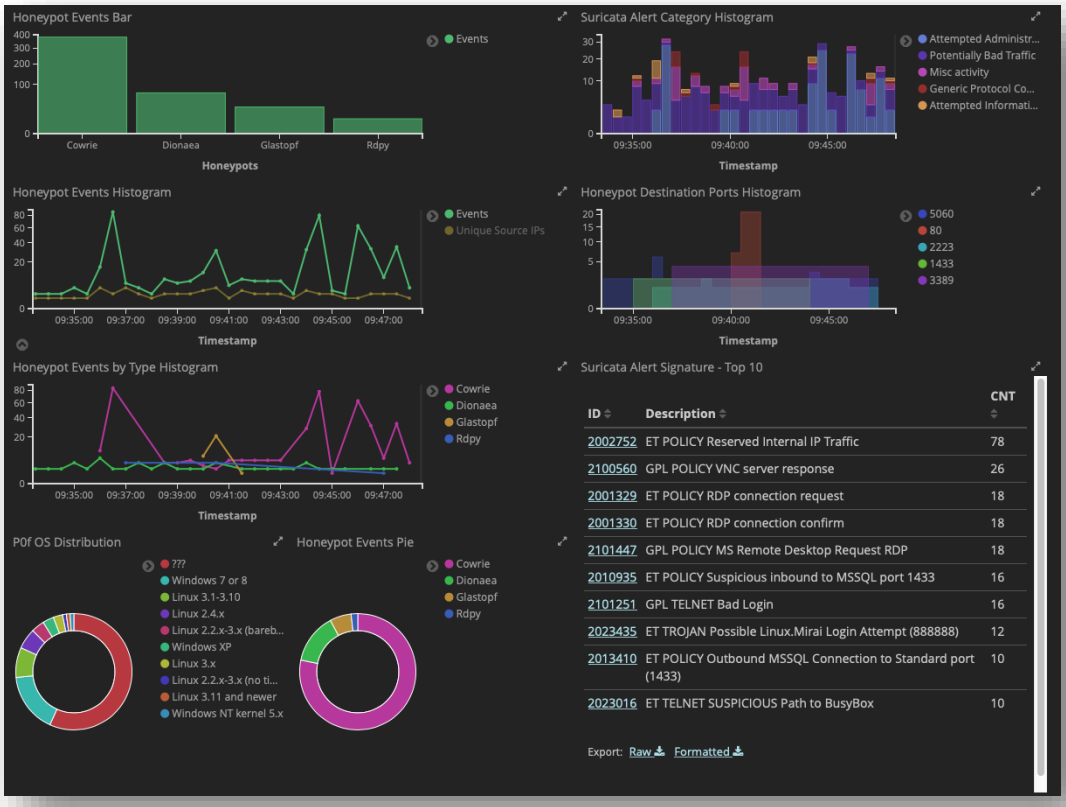

Figure 21: Visualization of results in HADES

5.1.10 Reliable-CTIs - Reliable Cyber-Threat intelligence sharing (UMU)

Name: Reliable-CTIs - Reliable Cyber-Threat intelligence sharing
Partner: UMU
Capability: Detect
Category (L2): Security Continuous Monitoring
Category (L3): Cyber Threat Intelligence
Type: Software: Component
Description: Enabler leveraging current Open Threat Intelligence platforms such as MISP, to share securely, trusted Cyber-Threat intelligence data between CERT/CSIRTS, companies and related entities. A multi-dimensional approach to quantify trust among involved stakeholders will devised, combining , for instance, the peer's reputation, the collaboration maturity, and the membership to federations. The trust model will drive the CTI secure data exchange.
Development Phase Status: Planning

Nowadays CSIRTs and CERTs are using Cyber-Threats Intelligence sharing platforms to exchange indicators of compromise, vector, and actors of cybersecurity attacks. The high-speed communication of this data offered by CTI sharing platforms helps to mitigate attacks by enabling systems to be reinforced before the attack occurs. This shifts to a proactive defense of shielding vulnerable and targeted systems and services.

Although being used more and more by more people, there is a lot of information that is not useful for some organizations or there are even cases that there is information that turns against them. This is one of the reasons why some kind of mechanism is needed to help trust the information obtained from the CTIs platform. This enabler will help to evaluate trustworthiness of the exchanged data as well as the organizations, sources and feeds used to receive Cyber Threat Intelligence information.

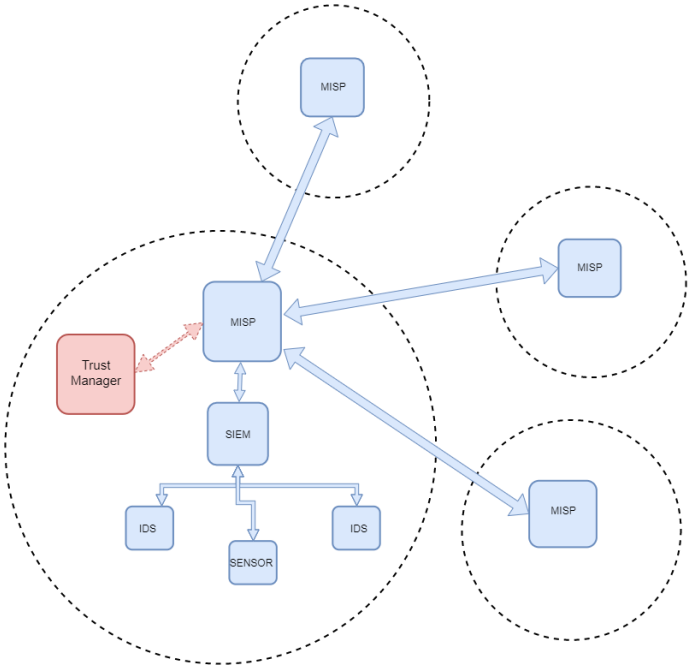

Figure 22: Structure Scheme of Reliable-CTIs enabler

To this aim, the *Reliable-CTIs* enabler will leverage existing open-source tools such as MISP to allow reliable CTI data sharing by relying on trust scores which are dynamically quantified by a Trust Manager following a multidimensional approach.

The scenario in Figure 22 represents a high-level view of the main components involved in the Reliable-CTI enabler. The figure represents several heterogeneous organizations, which means they do not share nature and may not have the same attack pattern, and each has a CTI sharing platform, such as MISP, to share information about cyberthreats and malware. Within the organization there is a SIEM (Security Event Manager), such as OSSIM, to obtain information from sensors and detectors of commitment indicators. There are also several sensors as well as IDS (Intrusion Detection System) and software to obtain information on systems, services, and networks. A Trust Manager component has been added to this scenario that will consider the information obtained from MISP in order to determine which exchanged data can be considered reliable and secure.

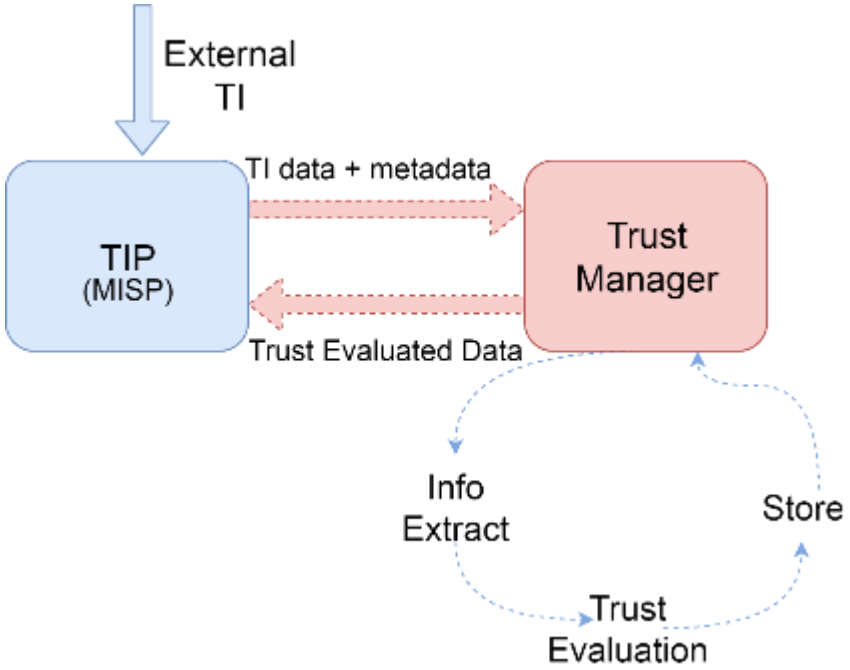

Figure 23: Trust Manager component

The Trust Manager component and its main interactions are depicted in Figure 23. As it can be seen, it communicates with the CTI platform (MISP) and uses the data obtained from the intelligence of the shared cyberthreat in order to quantify trustworthiness of the exchanged data. The Trust Manager will consider several factors, including the reputation and historical behaviour of the contributor (provider), the maturity of the collaboration, the organization of membership, the content of the shared data itself (e.g. Indicators of compromise), and context information and will assign trust values. After that it will quantify that information with a multidimensional approach, using for instance fuzzy logics, giving weights and values to each of these elements and additionally giving them a dimension. In this way, levels of trust will be created. The Trust Manager will store this trustworthiness information for future analysis, in order to improve the security mechanisms. Finally, the trust scores will be sent to TIP for be shown and to the

administrator and even to share the trust-score themselves with external entities. With regard to communications, they will be secure and non repudiable, since signatures will be used to encrypt the information.

These levels of confidence, proposed in the solution, can help Socs CERTs and CSISRTs to discern the information coming from the good CITs sharing systems that they can trust from that which is invented and act accordingly to this information. In addition, it can be used with other assets and tools proposed to grant to these systems of security of techniques of partial encryption, intelligence, and data extraction.

5.1.11 ENIDS: Edge Network Intrusion Detection System (UNITN/FBK)

Name: Edge Network Intrusion Detection System (ENIDS)
Partner: UNITN/FBK
Capability: Detect and Respond
Category (L2): Anomalies and Events, Mitigation
Category (L3): Intrusion Detection, DDoS protection
Type: Software: Component
Description: ENIDS implements an intrusion detection component based on the statistical properties of the network traffic (e.g., entropy values of header fields of packets). The output of the detection is used as input for a second component, a Linux kernel-based traffic filter that blocks all the packets classified as malicious by the detection component. ENIDS has been designed to work on resource-constrained systems such as the nodes of edge computing environments.
Development Phase Status: Implementation

With the recent trend of "network softwarisation", enabled by emerging technologies such as Network Function Virtualisation (NFV) and Software Defined Networking (SDN), system administrators of data centres and enterprise networks have started replacing dedicated hardware-based middleboxes with virtualised network functions running on commodity servers and end hosts. This radical change has facilitated the provisioning of advanced and flexible network services, ultimately helping system administrators and network operators to cope with the rapid changes on service requirements and networking workloads.

In this context, the Edge Network Intrusion Detection System (ENIDS) is a software solution for Linux-based host machines that combines traffic analysis and filtering components. The ENIDS architecture features a data plane composed of a set of programs that run on the host's kernel which are in charge of filtering malicious packets. Such programs are implemented with recent Linux kernel technologies such as the extended Berkeley Packet Filter (eBPF) and the eXpress Data Path (XDP). The intrusion detection logic is implemented in the user space of the host, where network traffic attributes are collected and processed to classify the traffic flows as either normal or malicious. The output of the detection process is used to program the traffic filtering policies in the kernel space. The overall architecture is depicted in Figure 24 and described below in the remainder of this section.

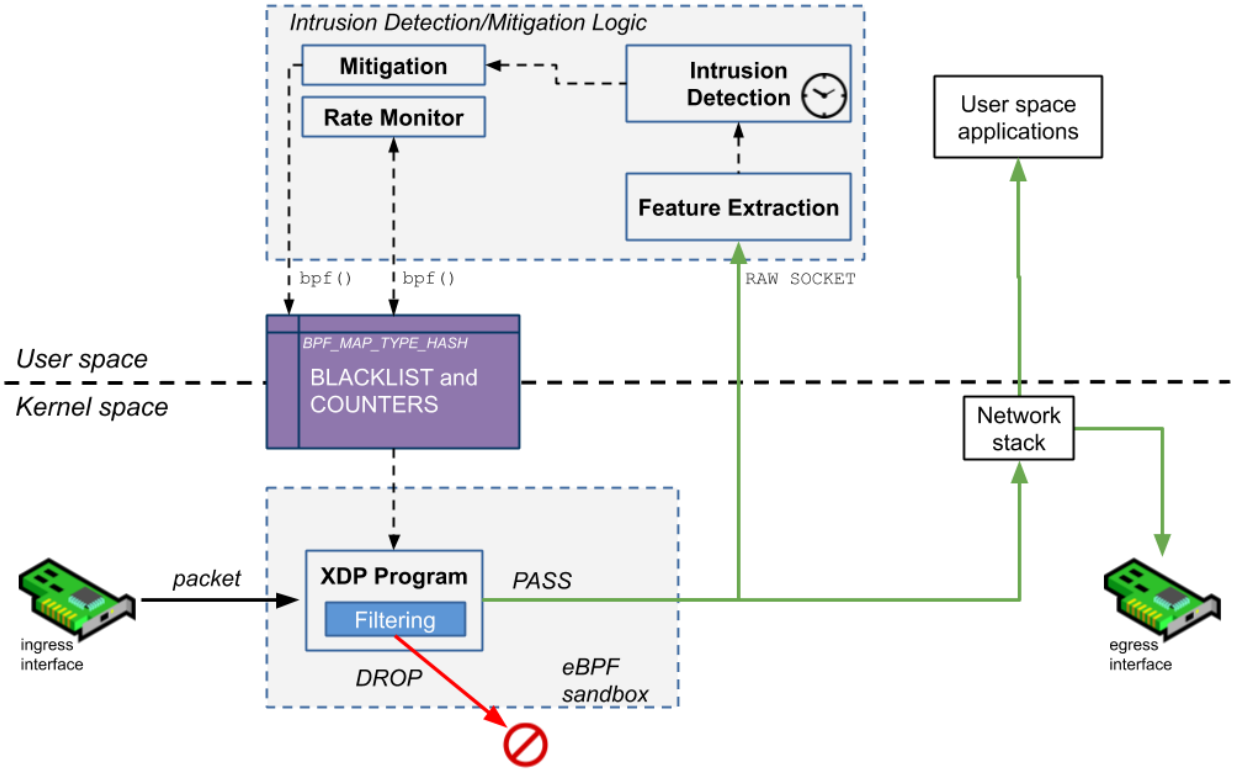

Figure 24: ENIDS architecture

Mitigation: The *XDP Filtering* program running in the kernel space matches the incoming packets against the content of a blacklist and drops them if the result is positive. Surviving packets are redirected to the *Network Stack* of operating system, which forwards it to the target user space applications or to the next hop in the path towards the final destination. The blacklist is implemented through an eBPF hash map that guarantees fast lookup operations.

Feature Extraction: The *Feature Extraction* module monitors the incoming traffic and collects relevant packet attributes required by the detection algorithm (e.g., IP addresses, protocols, flags, etc.). Being placed right after the mitigation module, it receives all the (presumed) benign traffic that has not been previously dropped.

Intrusion Detection: The current implementation of the Intrusion Detection module is based on an entropy-based algorithm for the detection of Distributed Denial of Service (DDoS) attacks. The algorithm operates on aggregated traffic statistics, and computes the Shannon entropy to detect variations in the distribution of traffic features observed in consecutive timeframes. The IP addresses of the sources of the DDoS attack are inserted into the blacklist by the *Mitigation* module and then used by the *Filtering* module for blocking the malicious traffic.

Rate Monitor: The blacklist is also used to keep track of the number of packets dropped for each malicious source stored in it. Such statistics, called Counters in Figure 24, are used by the user space program *Rate*

Monitor to remove from the Blacklist the sources that are no longer part of a DDoS attack, or that were erroneously classified as malicious by the Detection algorithm.

5.1.12 RoCe: Risk of Compromise estimation (UNITN)

Name: Risk of Compromise estimation of a given network
Partner: UNITN
Capability: Identify
Category (L2): Risk Assessment
Category (L3): Security Certification
Type: Algorithm
Description: RoCe is a methodology for estimation of a risk of compromise of a given network
Development Phase Status: Planning

The overall goal is to set up an experimental methodology that estimate the empirical hardness of exploiting the vulnerabilities in a network. Our key idea is to monitor the outcomes of Capture The Flag (CTF) to reproduce APTs attacks and estimate the risk of compromise. The experiment set-up would then follow the protocol:

1) Before the execution of any activity, subjects are given a questionnaire to collect information on their background and knowledge of attack techniques.

2) A scenario description is administered to subjects by either an individual reading or by an introductory presentation. Then, a training phase follows in which the expert in the testbed introduces its functionalities through a step-by-step tutorial.

3) The subjects apply their attacking skills to the scenario. In this case, a part of the design decisions would also include the presence (or absence) of defenders.

4) In this phase the outcome of the CTFs are analyzed to identify for example if and how a red team has successfully compromised the system. If an automatic assessment is not possible, then external evaluators assess the results of the CTFs providing also marks and comments. These expert evaluators should possibly be the experimenters but rather external experts contracted for the purpose. If human defenders are included there should be an assessment also of their activities to be used as a controlling factor of the possible mitigation effect that this might have had. It is important to underline that the external evaluators determine the outcome of the exercises and do not assess the risk of the network.

5) A post-task questionnaire is administered to the subjects to gather their perception of tools they used and on the experiment as a whole.

To estimate the risk of compromise, the factor of interest is the likelihood of compromise. A simple measure might be the number of teams that were able to successfully attack (within the duration of the experiment) divided by the total number of teams in the experiment, assuming all teams were at the same skill level and have access to the same tools. However, this would suffer from several limitations in terms of transferability of the results. Another possibility is to define the likelihood as the inverse of the time to compromise from

the moment in which the attack started to the moment in which a pre-defined compromise is achieved (i.e. attacked network's logs reveal the malicious behavior).

To reproduce the characteristics of the APTs, we need to assess the skills of the participants of the CTF to choose the right set of skills for the simulation in the CTF. The assessment of the attacker skills strictly depends on the information collected from the threat intelligence. The knowledge about which threat actors are active in the specific fields and their procedures (TTPs) allows to identify the skills needed to emulate these attacks. Defender skills can be assessed in the same way. Additionally, the NIST's NICE Framework can be used to identify skills and abilities required for specific roles in the Blue team. The assessment can be done through a self-assessment validated with some technical questions.

5.2 High-level architectural overview

Figure 25 below highlights the role of 'Task 3.4 Security Intelligence' in the overall CyberSec4Europe architecture. Its main responsibilities reside in the intelligence plane, as can be expected.

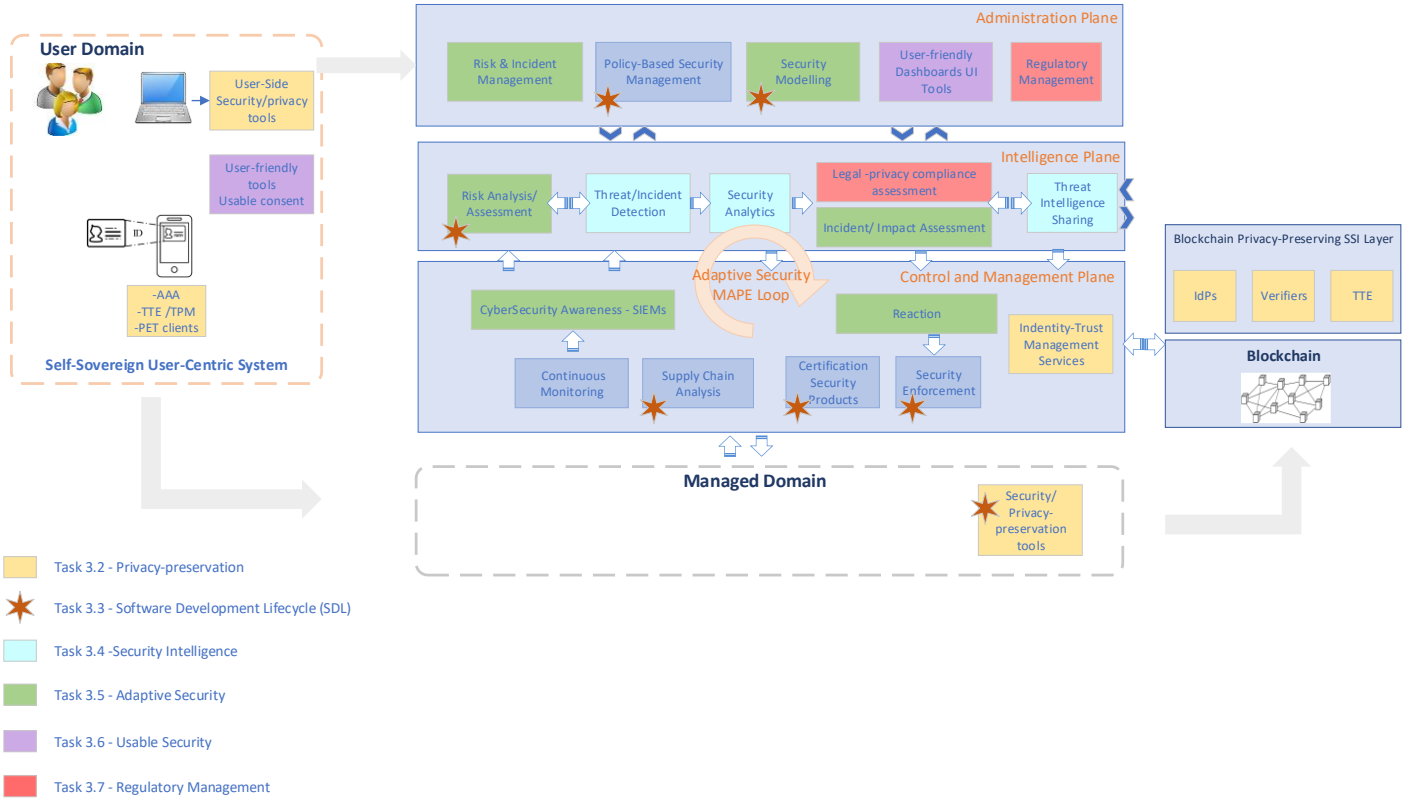

Figure 25: CyberSec4Europe architecture and building blocks per WP3 task [Deliverable D3.1]

The 3 categories in which the above assets play a role, include:

1. **Threat/incident detection:** TIE, Briareos, UASD, EBIDS, IntelFrame, ENIDS
2. **Security analytics:** TIE, EBIDS, IntelFrame, NetGen, JUDAS, HADES, ENIDS, RoCe
3. **Threat Intelligence sharing:** TIE, TATIS, Reliable-CTIs

6 Research challenges and requirements

This section reviews the requirements elicited in deliverable D5.1 [D51 2019], and highlights to what extent to software assets of Task 3.4 and identified in the previous section are able to address these requirements.

6.1 Asset mapping on WP5 requirements

Vertical	Requirements addressed
Open Banking	
Supply Chain Security Assurance	
Privacy-Preserving Identity Management	
Incident Reporting	IR-F05, IR-F10, IR-F25, IR-SP01, IR-SP02, IR-SP03, IR-SP04, IR-LF01, IR-U01, IR-OP01, IR-MP02
Maritime Transport	
Medical Data Exchange	
Smart Cities	

Table 2: Requirements mapping for TIE (ATOS)

Vertical	Requirements addressed
Open Banking	OB-SP14, OB-SP15, OB-SP20, OB-LF03, OB-LF04, OB-LR05, OB-LR06
Supply Chain Security Assurance	SCH-LR02
Privacy-Preserving Identity Management	IDM-SP04
Incident Reporting	
Maritime Transport	
Medical Data Exchange	
Smart Cities	SMC-F02, SMC-SP06, SMC-SP07

Table 3: Requirements mapping for Briareos (C3P)

Vertical	Requirements addressed
Open Banking	OB-LR02
Supply Chain Security Assurance	
Privacy-Preserving Identity Management	
Incident Reporting	
Maritime Transport	
Medical Data Exchange	
Smart Cities	SMC-F17

Table 4: Requirements mapping for UASD (CNR)

Vertical	Requirements addressed
Open Banking	OB-U01, OB-SP15, OB-SP20, OB-LF04,OB-LR05
Supply Chain Security Assurance	
Privacy-Preserving Identity Management	
Incident Reporting	
Maritime Transport	
Medical Data Exchange	
Smart Cities	

Table 5: Requirements mapping for EBIDS (CNR)

Vertical	Requirements addressed
Open Banking	
Supply Chain Security Assurance	
Privacy-Preserving Identity Management	
Incident Reporting	IR-F04, IR-F05, IR-F13, IR-F17, IR-F19, IR-F24, IR-F25, IR-U03
Maritime Transport	

Medical Data Exchange	
Smart Cities	

Table 6: Requirements mapping for IntelFrame (DTU)

Vertical	Requirements addressed
Open Banking	
Supply Chain Security Assurance	
Privacy-Preserving Identity Management	
Incident Reporting	IR-SP01, IR-SP02, IR-SP03, IR-SP04, IR-LF01, IR-OP01
Maritime Transport	
Medical Data Exchange	
Smart Cities	

Table 7: Requirements mapping for TATIS (KUL)

Vertical	Requirements addressed
Open Banking	OB-SP15, OB-SP23
Supply Chain Security Assurance	SCH-SP07
Privacy-Preserving Identity Management	IDM-SP06
Incident Reporting	IR-F04
Maritime Transport	
Medical Data Exchange	MD-SP02
Smart Cities	SMC-SP07

Table 8: Requirements mapping for NetGen (POLITO)

Vertical	Requirements addressed
Open Banking	

Supply Chain Security Assurance	
Privacy-Preserving Identity Management	
Incident Reporting	IR-F02, IR-F03, IR-F15, IR-F19, SMC-F07, SMC-F09, SMC-F15, SMC-SP04, SMC-SP11
Maritime Transport	
Medical Data Exchange	
Smart Cities	

Table 9: Requirements mapping for JUDAS (UMA)

Vertical	Requirements addressed
Open Banking	
Supply Chain Security Assurance	
Privacy-Preserving Identity Management	
Incident Reporting	IR-F04, IR-F05, IR-F17, IR-F18, IR-F19, IR-F24, IR-F25
Maritime Transport	
Medical Data Exchange	
Smart Cities	

Table 10: Requirements mapping for HADES (UMA)

Vertical	Requirements addressed
Open Banking	
Supply Chain Security Assurance	
Privacy-Preserving Identity Management	
Incident Reporting	IR-F11, IR-F10, IR-SP01, IR-SP02, IR-SP04, IR-LF01, IR-OP01
Maritime Transport	
Medical Data Exchange	

Smart Cities	

Table 11: Requirements mapping for Reliable-CTIs (UMU)

Vertical	Requirements addressed
Open Banking	
Supply Chain Security Assurance	
Privacy-Preserving Identity Management	
Incident Reporting	
Maritime Transport	
Medical Data Exchange	
Smart Cities	SMC-SP04, SMC-SP08

Table 12: Requirements mapping for ENIDS (UNITN/FBK)

Vertical	Requirements addressed
Open Banking	OB-SP07, OB-SP08, OB-SP09, OB-SP10, OB-SP13, OB-SP14, OB-SP15, OB-SP20, OB-SP21, OB-SP23, OB-LF03, OB-LF04, OB-LF05
Supply Chain Security Assurance	
Privacy-Preserving Identity Management	
Incident Reporting	
Maritime Transport	
Medical Data Exchange	MD-SP07
Smart Cities	SMC-SP01, SMC-SP02, SMC-SP03, SMC-SP04

Table 13: Requirements mapping for RoCe (UNITN)

6.2 Gap analysis and research challenges

With respect to the vertical domains at hand, there is a clear link with the `Incident Reporting' demonstration case. However, a variety of assets are fairly generic and may be tuned to also address the needs of other use

cases. Specifically in terms of the AI powered security analytics asset, the availability of datasets and insight knowledge of relevant threats may help to validate these solutions.

Another observation is that the many assets described in the document are at different levels of maturity. Some are still in the planning stage, while others will be ready for testing in the second year of the project. Conceptually, some of them target similar goals (e.g. intrusion detection), and further analysis and research will be necessary not only to identify how they may complement one another.

Privacy and compliance in the frame of threat intelligence and the management of digital evidence is a concern that has been discussed in the previous sections. Thus far, there are no assets available that allow for the notification of incident while being compliant with a variety of regulations (e.g. GDPR) and directives (e.g. PSD2). Various other vertical domains expressed an interest in Big Data-like solutions that allow for privacy preserving data analytics for purposes beyond threat intelligence. Given the complexity of the algorithms involved and the continuous emergence of side channel attacks, it is expected that the arms race between attackers and defenders will continue, making it more challenging to map compliance requirements onto enabling technologies.

Last but not least, the overall architecture depicts how security intelligence is subdivided into 3 building blocks. However, the translation of a conceptual overview into a technical realization and integration – within this task but also beyond – is an action point for the future. Most likely, this will not result in a single unified platform, but rather in multiple instances of the platform in which a subset of assets are integrated, so that they can be used in the frame of certain demonstration cases (WP5) but also for further research (WP3).

7 Conclusion

This deliverable reports the results and outcomes of Task 3.4 on Security Intelligence. It presents a first list on the requirements and challenges to manage digital evidence, and provides a high-level overview of the relevant state-of-the-art in the area of cyber threat intelligence, the management and sharing of incident information, the use and limitations of machine learning, and the implications on privacy.

Additionally, this document lists relevant components, algorithms, and software building blocks of the project partners that can help address these requirements. Further research and development is necessary as these assets are at different levels of maturity. Also, the integration of these assets across the different tasks of WP3 will be an activity that will take place throughout the duration of the project.

The goal of cyber threat intelligence to prevent an attack or shorten the window between compromise and detection, and this on evidence-based knowledge. To achieve this objective, we are faced with challenges from a human and technological nature:

- Lack of trust in the way threat intelligence information is handled by receiving parties is a key factor why organizations are reluctant to share information
- The quality (rather than the quantity) of threat feeds and events must increase for a reliable and automated threat analysis and mitigation
- The event based sharing philosophy of threat intelligence platforms does not match well with data driven and AI powered threat intelligence
- The application of security techniques – such as end-to-end encryption, onion routing, etc. – make it harder to harvest threat intelligence from monitoring data and event logs
- The AI capabilities of contemporary threat intelligence platforms enable new kinds of attacks that allow adversaries to learn how to evade detection
- Machine learning models that underpin cybersecurity solutions may leak sensitive information, and need strong protection to avoid privacy concerns or loss of reputation

While individual steps have been made, it is clear that a holistic solution is not trivial. These research challenges and high-level requirements will be the main drivers to enhance existing assets and develop new ones within the frame of Task 3.4 to bridge the gap with the current state-of-practice and to increase the technological readiness towards a first set of demonstrators in WP5.

8 References

[Abadi 2016] Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., et al. (2016). "Deep learning with differential privacy," in Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (New York, NY: ACM), 308–318. doi: 10.1145/2976749.2978318

[Acien 2018] A. Acien, A. Nieto, G. Fernandez, and J. Lopez, A comprehensive methodology for deploying IoT honeypots, 15th International Conference on Trust, Privacy and Security in Digital Business (TrustBus 2018), vol. LNCS 11033, Springer Nature Switzerland AG, pp. 229–243, 09/2018

[Altman 2018] Altman, Micah, Alexandra B. Wood, David O'Brien, and Urs Gasser. "Practical approaches to big data privacy over time." (2018).

[Bao 2018] Bao, R, Chen, Z, Obaidat, MS. Challenges and techniques in Big data security and privacy: A review. Security and Privacy 2018; 1:e13. https://doi.org/10.1002/spy2.13

[Barreno 2010] Barreno, M., Nelson, B., Joseph, A.D. and Tygar, J.D., 2010. The security of machine learning. Machine Learning, 81(2), pp.121-148.

[Biggio 2012] Battista Biggio, Blaine Nelson, and Pavel Laskov. 2012. Poisoning attacks against support vector machines. In Proceedings of the 29th International Conference on International Conference on Machine Learning (ICML'12). Omnipress, USA, 1467-1474.

[Bonawitz 2017] Keith Bonawitz, Vladimir Ivanov, Ben Kreuter, Antonio Marcedone, H. Brendan McMahan, Sarvar Patel, Daniel Ramage, Aaron Segal, and Karn Seth. 2017. Practical Secure Aggregation for Privacy-Preserving Machine Learning. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS '17). ACM, New York, NY, USA, 1175-1191. DOI: https://doi.org/10.1145/3133956.313398

[Briland 2017] Briland Hitaj, Giuseppe Ateniese, and Fernando Perez-Cruz. 2017. Deep Models Under the GAN: Information Leakage from Collaborative Deep Learning. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security (CCS '17). ACM, New York, NY, USA, 603-618. DOI: https://doi.org/10.1145/3133956.3134012

[Bro 2017] The Bro Project. Bro introduction. Website. https://www.bro.org/sphinx/, August 2017.

[Bromiley 2016] Bromiley, M., 2016. Threat intelligence: What it is, and how to use it effectively. SANS Institute InfoSec Reading Room, 15.

[Buczak 2015] Buczak, Anna & Guven, Erhan. (2015). A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection. IEEE Communications Surveys & Tutorials. 18. 1-1. 10.1109/COMST.2015.2494502.

[Burger 2014] Eric W. Burger, Michael D. Goodman, Panos Kampanakis, and Kevin A. Zhu. 2014. Taxonomy Model for Cyber Threat Intelligence Information Exchange Technologies. In Proceedings of the 2014 ACM Workshop on Information Sharing & Collaborative Security (WISCS '14). ACM, New York, NY, USA, 51-60. DOI: https://doi.org/10.1145/2663876.2663883

[Chantzios 2019] Chantzios, T., Koloveas, P., Skiadopoulos, S., Kolokotronis, N., Tryfonopoulos, C., Bilali, V.G. and Kavallieros, D., The quest for the appropriate cyber-threat intelligence sharing platform. In 8 th International Conference on Data Science, Technology and Applications (DATA 2019) (pp. 26-28).

[Chen 2018] S. Chen, M. Xue, L. Fan, S. Hao, L. Xu, H. Zhu, B. Li. Automated Poisoning Attacks and Defenses in Malware Detection Systems: An Adversarial Machine Learning Approach. Elsevier Computers & Security. Volume 73, March 2018, Pages 326-344.

[Conti 2018] Conti M., Dargahi T., Dehghantanha A. (2018) Cyber Threat Intelligence: Challenges and Opportunities. In: Dehghantanha A., Conti M., Dargahi T. (eds) Cyber Threat Intelligence. Advances in Information Security, vol 70. Springer, Cham, https://doi.org/10.1007/978-3-319-73951-9_1

[D31 2019] Skarmeta A. et al. Cyber Security for Europe - Deliverable D3.1: Common Framework Handbook 1, October 2019, https://cybersec4europe.eu/publications/deliverables/

[D41 2019] Ferreira A. et al. Cyber Security for Europe - Deliverable D4.1: Requirements Analysis from Vertical Stakeholders, July 2019, https://cybersec4europe.eu/publications/deliverables/

[D51 2019] Sforzin A. et al. Cyber Security for Europe - Deliverable D5.1: Requirements Analysis of Demonstration Cases, May 2019, https://cybersec4europe.eu/publications/deliverables/

[Ding 2018] Ding, Q., Li, Z., Haeri, S., Trajković, L.: Application of machine learning techniques to detecting anomalies in communication networks: Datasets and feature selection algorithms. In: M. Conti, A. Dehghantanha, T. Dargahi (eds.) Cyber Threat Intelligence, chap. 3, p. in press. Springer - Advances in Information Security series (2018)

[DiSIEM 2018] DiSIEM Consortium. OSINT data fusion and analysis architecture. DiSIEM Project Deliverable 4.2. March 2018.

[Elastic 2016] Elasticsearch BV. Elastic. Website. https://www.elastic.co, November 2016.

[Emsisoft 2019] EmsiSoft Malware Lab. State of Ransomware in the U.S.: 2019 Report for Q1 to Q3. https://blog.emsisoft.com/en/34193/state-of-ransomware-in-the-u-s-2019-report-for-q1-to-q3/, (October 2019)

[ENISA 2018] ENISA, Limits of TISPs, 2018, https://www.enisa.europa.eu/publications/exploring-the-opportunities-and-limitations-of-current-threat-intelligence-platforms/at_download/fullReport

[ENISA 2019] ENISA, Considerations on the Traffic Light Protocol, https://www.enisa.europa.eu/topics/csirts-in-europe/glossary/considerations-on-the-traffic-light-protocol, Last visited: October 1, 2019

[Faiella 2019] M. Faiella, G. Gonzalez-Granadillo, I. Medeiros, R. Azevedo, S. Gonzalez-Zarzosa. Enriching Threat Intelligence Platforms Capabilities. In proceedings of the 16th International Conference on Security and Cryptography (SECRYPT) 2019, July 2019.

[Fekolkin 2015] Roman Fekolkin. Intrusion Detection and Prevention Systems: Overview of Snort and Suricata. 2015.

[Gardiner 2016] Gardiner, J. and Nagaraja, S., 2016. On the security of machine learning in malware c&c detection: A survey. ACM Computing Surveys (CSUR), 49(3), p.59.

[Gonzalez-Granadillo 2019] G. Gonzalez-Granadillo, M. Faiella, I. Medeiros, R. Azevedo, S. Gonzalez-Zarzosa; Enhancing Information Sharing and Visualization Capabilities in Security Data Analytic Platforms; In the Dependable Systems and Networks (DSN) conference workshop on Data-Centric Dependability and Security (DCDS) 2019, April 2019.

[Gruschka 2018] Gruschka, Nils, Vasileios Mavroeidis, Kamer Vishi, and Meiko Jensen. "Privacy Issues and Data Protection in Big Data: A Case Study Analysis under GDPR." In 2018 IEEE International Conference on Big Data (Big Data), pp. 5027-5033. IEEE, 2018.

[Homayoun 2018] Homayoun, S., Ahmadzadeh, M., Hashemi, S., Dehghantanha, A., Khayami, R.: BoTShark: A deep learning approach for botnet traffic detection. In: M. Conti, A. Dehghantanha, T. Dargahi (eds.) Cyber Threat Intelligence, chap. 7, p. in press. Springer - Advances in Information Security series (2018)

[Jiang 2016] Jiang, H., Nagra, J. and Ahammad, P., 2016. Sok: Applying machine learning in security-a survey. arXiv preprint arXiv:1611.03186.

[Kantarcioglu 2019] Kantarcioglu Murat, Ferrari Elena, Research Challenges at the Intersection of Big Data, Security and Privacy, Frontiers in Big Data, Vol. 2, 2019, ISSN 2624-909X, DOI 10.3389/fdata.2019.00001, https://www.frontiersin.org/article/10.3389/fdata.2019.00001

[Kesarwani 2018] Manish Kesarwani, Bhaskar Mukhoty, Vijay Arya, and Sameep Mehta. 2018. Model Extraction Warning in MLaaS Paradigm. In Proceedings of the 34th Annual Computer Security Applications Conference (ACSAC '18). ACM, New York, NY, USA, 371-380. DOI: https://doi.org/10.1145/3274694.3274740

[Khalil 2015] G Khalil. Open source ids high performance shootout. February 2015.

[Laud 2015] Laud P, Kamm L. Practical applications of secure multiparty computation. Appl Secure Multiparty Comput. 2015;13:246.

[Mahdavifar 2019] Samaneh Mahdavifar, Ali A. Ghorbani, Application of deep learning to cybersecurity: A survey, Neurocomputing, Volume 347, 2019, Pages 149-176, ISSN 0925-2312, https://doi.org/10.1016/j.neucom.2019.02.056.

[Mahmood 2013] Mahmood, T. and Afzal, U., 2013, December. Security analytics: Big data analytics for cybersecurity: A review of trends, techniques and tools. In 2013 2nd national conference on Information assurance (ncia) (pp. 129-134). IEEE.

[McDaniel 2016] McDaniel, P., Papernot, N. and Celik, Z.B., 2016. Machine learning in adversarial settings. IEEE Security & Privacy, 14(3), pp.68-72.

[McMillan 2013] McMillan R. Definition: threat intelligence. Gartner; 2013. https://www.gartner.com/en/documents/2487216/definition-threat-intelligence

[Menges 2018] Florian Menges, Günther Pernul, A comparative analysis of incident reporting formats, Computers & Security, Vol. 73, 2018, pp. 87-101, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2017.10.009

[Nasr 2018] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2018. Machine Learning with Membership Privacy using Adversarial Regularization. In Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security (CCS '18). ACM, New York, NY, USA, 634-646. DOI: https://doi.org/10.1145/3243734.3243855

[Nelson 2017] Nelson B, Olovsson T. Security and privacy for Big data: a systematic literature review. IEEE International Conference on Big Data. IEEE, 2017:3693-3702.

[Nieto 2019] A. Nieto, R. Rios, and J. Lopez, "Privacy-Aware Digital Forensics", Security and Privacy for Big Data, Cloud Computing and Applications, Lizhe Wang, Wei Ren, Raymoond Choo and Fatos Xhafa, The Institution of Engineering and Technology (IET) , 09/2019.

[OISF 2015] The Open Information Security Foundation. Suricata tutorial. Presentation. https://resources.sei.cmu.edu/asset_files/Presentation/2016_017_001_449890.pdf, December 2015.

[Park 2017] Park, W. and Ahn, S., 2017. Performance comparison and detection analysis in snort and suricata environment. Wireless Personal Communications, 94(2), pp.241-252.

[Phong 2018] Phong, Le Trieu, Yoshinori Aono, Takuya Hayashi, Lihua Wang, and Shiho Moriai. "Privacy-preserving deep learning via additively homomorphic encryption." IEEE Transactions on Information Forensics and Security 13, no. 5 (2018): 1333-1345.

[Prasser 2015] Prasser, Fabian, and Florian Kohlmayer. "Putting statistical disclosure control into practice: The ARX data anonymization tool." In Medical Data Privacy Handbook, pp. 111-148. Springer, Cham, 2015.

[Qamar 2017] Qamar, S., Anwar, Z., Rahman, M.A., Al-Shaer, E., Chu, B.T.: Data-driven analytics for cyber-threat intelligence and information sharing. Comput. Secur. 67(C), 35-58 (Jun 2017)

[Roesch 1999] Roesch, M., 1999, November. Snort: Lightweight intrusion detection for networks. In Lisa (Vol. 99, No. 1, pp. 229-238).

[Rubinstein 2009] Benjamin I.P. Rubinstein, Blaine Nelson, Ling Huang, Anthony D. Joseph, Shing-hon Lau, Satish Rao, Nina Taft, and J. D. Tygar. 2009. ANTIDOTE: understanding and defending against poisoning of anomaly detectors. In Proceedings of the 9th ACM SIGCOMM conference on Internet measurement (IMC '09). ACM, New York, NY, USA, 1-14

[Sauerwein 2017] Clemens Sauerwein, Christian Sillaber, Andrea Mussmann, Ruth Breu: Threat Intelligence Sharing Platforms: An Exploratory Study of Software Vendors and Research Perspectives. Wirtschaftsinformatik 2017

[Schreiber 2016] Joe Schreiber. Open-source IDS tools overview. Website. https://www.alienvault.com/blogs/security-essentials/open-source-intrusion-detection-tools-a-quick-overview, November 2016.

[Smith 2017] Travis Smith. Integrating bro ids with the elastic stack. Website. https://www.elastic.co/blog/bro-ids-elastic-stack, August 2017.

[Snort 2017] Snort Team. Snort overview. Website. http://manual-snort-org.s3-website-us-east-1.amazonaws.com/node2.html, January 2017.

[Sommer 2003] Robin Sommer. Bro: An open source network intrusion detection system. In DFN-Arbeitstagung über Kommunikationsnetze, pages 273–288, 2003.

[Stanger 2015] James Stanger. Detecting intruders with suricata. Website. http://www.admin-magazine.com/Archive/2015/27/Detecting-intruders-with-Suricata, June 2015.

[Suricata 2016] Suricata. Suricata. Website. https://suricata-ids.org, November 2016.

[Ten 2010] Ten, C.W., Manimaran, G. and Liu, C.C., 2010. Cybersecurity for critical infrastructures: Attack and defense modeling. IEEE Transactions on Systems, Man, and Cybernetics-Part A: Systems and Humans, 40(4), pp.853-865.

[Tounsi 2018] Wiem Tounsi, Helmi Rais, A survey on technical threat intelligence in the age of sophisticated cyber attacks, Computers & Security, Volume 72, 2018, Pages 212-233, ISSN 0167-4048, https://doi.org/10.1016/j.cose.2017.09.001.

[Tramèr 2016] Florian Tramèr, Fan Zhang, Ari Juels, Michael K. Reiter, and Thomas Ristenpart. 2016. Stealing machine learning models via prediction APIs. In Proceedings of the 25th USENIX Conference on Security Symposium (SEC'16), Thorsten Holz and Stefan Savage (Eds.). USENIX Association, Berkeley, CA, USA, 601-618.

[Turner 2016] Mav Turner, Building stronger defences through sharing, Computer Fraud & Security, Vol. 2016, Issue 9, 2016, pp. 11-14, ISSN 1361-3723, https://doi.org/10.1016/S1361-3723(16)30071-9.

[van de Kamp 2015] van de Kamp, T., Peter, A., Everts, M. H., & Jonker, W. (2016, October). Private Sharing of IOCs and Sightings. In Proceedings of the 2016 ACM on Workshop on Information Sharing and Collaborative Security (pp. 35-38). ACM.

[Verizon 2019] Verizon: 2019 Data Breach Investigations Report. Computer Fraud & Security, Vol 2019, Iss 6, June 2019, page 4, https://doi.org/10.1016/S1361-3723(19)30060-0, https://enterprise.verizon.com/resources/reports/2019-data-breach-investigations-report.pdf (Jun 2019)

[Verma 2018] Rakesh Verma. 2018. Security Analytics: Adapting Data Science for Security Challenges. In Proceedings of the Fourth ACM International Workshop on Security and Privacy Analytics (IWSPA '18). ACM, New York, NY, USA, 40-41.

[Wagner 2016] Cynthia Wagner, Alexandre Dulaunoy, Gérard Wagener, and Andras Iklody. 2016. MISP: The Design and Implementation of a Collaborative Threat Intelligence Sharing Platform. In Proceedings of

the 2016 ACM on Workshop on Information Sharing and Collaborative Security (WISCS '16). ACM, New York, NY, USA, 49-56. DOI: https://doi.org/10.1145/2994539.2994542

[White 2013] Joshua S White, Thomas Fitzsimmons, and Jeanna N Matthews. Quantitative analysis of intrusion detection systems: Snort and suricata. In SPIE Defense, Security, and Sensing, pages 875704–875704. International Society for Optics and Photonics, 2013.