Advanced Cyber-Threat Intelligence, Detection, and Mitigation Platform for a Trusted Internet of Things
Grant Agreement: 786698

D5.2 Cyber-threat intelligence sharing

Work Package 5: Key proactive technologies and cyber-threat intelligence

Document Dissemination Level

P	Public	☒
CO	Confidential, only for members of the Consortium (including the Commission Services)	☐

Document Due Date: 31/01/2020
Document Submission Date: 31/01/2020

Co-funded by the Horizon 2020 Framework Programme of the European Union

Document Information

Deliverable number:	**5.2**
Deliverable title:	Cyber-threat intelligence sharing
Deliverable version:	1.0
Work Package number:	WP5
Work Package title:	Key Proactive technologies and cyber-threat intelligence
Due Date of delivery:	31/01/2020
Actual date of delivery:	31/01/2020
Dissemination level:	Public
Editor(s):	Dimitrios Kavallieros, Vasiliki-Georgia Bilali, George Kokkinis, Athanasios Grigoriadis (KEMEA)
Contributor(s):	Spiros Skiadopoulos, Thanasis Chantzios, Nicholas Kolokotronis (UOP) Gueltoum Bendiab, Bogdan Ghita (CSCAN) Stefano Cuomo, Simone Naldini (MATHEMA)
Reviewer(s):	Olga Gkotsopoulou, Paul Quinn (VUB) Gohar Sargsyan (CGI)
Project name:	Advanced Cyber-Threat Intelligence, Detection, and Mitigation Platform for a Trusted Internet of Things
Project Acronym	Cyber-Trust
Project starting date:	1/5/2018
Project duration:	36 months
Rights:	Cyber-Trust Consortium

Version History

Version	Date	Beneficiary	Description
0.1	13.11.2019	KEMEA	ToC
0.2	2.12.2019	KEMEA	Section 1, Section 2
0.3	13.01.2019	KEMEA	Section 3.1 ,3.2, 3.3, 3.4, 3.5
0.4	20.01.2019	UOP	Section 3.3
0.5	22.01.2019	KEMEA, UOP, CSCAN	Section 4, Section 6
0.6	24.01.2019	MATHEMA	Section 5
0.7	24.01.2019	KEMEA	Conclusion
0.8	29.01.2020	CGI, VUB	Review
0.9	30.01.2020	KEMEA	Final version for 2nd review
1.0	31.01.2020	KEMEA	Final version for submission

Acronyms

ACRONYM	EXPLANATION
A	Actor
API	Application Programming Interface
AS	Autonomous System
ASN	Autonomous System Number
BTC	Bitcoin
CEF	Common Event Format
CIDR	Classes Inter-Domain Routing
CPE	Common Platform Enumeration
CT	Cyber-Trust
CTI	Cyber-Threat Intelligence
CVE	Common Vulnerabilities and Exposures
CWE	Common Weakness Enumeration
D	Deliverable
DB	Database
DNS	Domain Name System
DPIA	Data Protection Impact Assessment
EQL	Event Query Language
evDB	Enriched Vulnerability Database
FR	Functional Requirement
HTTP	Hypertext Transfer Protocol
ID	Identification
IDS	Intrusion Detection System
IoC	Indicator of Compromise
IoT	Internet of Things
IP	Internet Protocol
ISP	Internet Service Provider
JSON	JavaScript Object Notation
LEA	Law Enforcement Agency
M	Month
MISP	Malware Information Sharing Platform
NFR	Non-Functional Requirement
OCR	Optical Character Recognition
PHP	Hypertext Pre-processor
REST	Representational State Transfer
SHO	Smart Home Owner
SQL	Structured Query Language

SSL	Secure Socket Layers
STIX	Structured Threat Information Expression
T	Task
TMS	Trust Management Service
UCG	Use Case Scenario
UI	User Interface
URL	Uniform Resource Locator
VERIS	Vocabulary for Event Recording and Incident Sharing
WP	Work Package
XML	Extensible Markup Language

Contents

Table of Figures

Table of Tables

1. Executive summary

Since the quantity of software vulnerabilities and malicious attack techniques have over exceeded every feasible limit the latest years, Cyber-threat Intelligence (CTI) sharing is very crucial tool for every organisation and technological application platform (e.g. Internet of Things (IoT) applications). Cyber-threat intelligence is any information that can help an organisation identify, assess, monitor, and respond to cyber threats. Examples of such information include indicators (system artifacts or observables associated with an attack), security alerts, threat intelligence reports, as well as recommended security tool configurations [1],[2].

Information sharing gives the opportunity to organisations and internet of things (IoT) applications to defend themselves. Proactive information-sharing takes under consideration known attacks, various defensive strategies and defensive mitigations in order to build resilience across cyber-tools and organisations participating within a given trust community, creating herd immunity against targeted attackers and threats that others have seen within their own networks [3].

Assuming a cyber-threat scale that measures the frequency of cyber threats in a corporate environment, data breaches reach the top of this ladder, something that subsequently costs a lot to businesses. For this reason, every year more effort and money are invested from businesses to create a robust cyber security framework. According to Experian's Sixth Annual Study: “Is Your Company Ready for a Big Data Breach”, only the 36% of businesses have taken the right measures in order to tackle and overcome a data breach [4].

Due to the above fundamental reasons, emphasis is given to the cyber-threat sharing service functionalities within the project. The component that is responsible for that is the Sharing Service component. It is consisted by two (2) parts,

a) a sharing platform and

b) a language format that is used as a sharing mechanism towards information transmission

both were selected in a previous deliverable of Cyber-Trust (see Section 3). For the former part of the Sharing Service was selected *Malware Information Sharing Platform (MISP)* and for the latter part was selected *Structured Threat Information Expression (STIX).*

MISP capabilities and specifications are described below together with a vast amount of instructions and terminologies. A part of these functions is installed and operates in the sharing service and another part determined by the administrator and the users.

2. Introduction

This deliverable gives detailed information for the cyber-threat intelligence sharing component [A07+A09] of Cyber-Trust "Advanced Cyber-Threat Intelligence, Detection and Mitigation Platform for a Trusted Internet of Things". This deliverable (D5.2 "Cyber-Threat Intelligence Sharing") is based on the knowledge that has already been acquired from previous documents. Such information varies, regarding the use-case scenarios, end-user requirements, Cyber-Trust components specifications as well as the Cyber-Trust architecture. The knowledge of the submitted deliverables will become the basis on which the D5.2 will provide enriched and advanced information for the cyber-threat intelligence sharing component of Cyber-Trust and will form a unique and coherent document.

2.1 Purpose of the document

The main purpose of this deliverable (D5.2 "Cyber-Threat Intelligence Sharing") is to introduce the reader to the Sharing Service [A07+A09] and to illustrate its role within the Cyber-Trust project. Moreover, to imprint the CTI capabilities of the sharing platform (MISP), as well as to present the MISP documentation for users' common understanding. Some CTI capabilities range from the identification and alerting of a simple cyber-threat to the analysis of information about the intent, opportunities of adversaries in cyberspace.

For the excellent understanding of the Cyber-Trust platform, in the following sections, we will present:

- the application architecture of the MISP platform,
- the usage of MISP,

the sources and information stored in MISP,

- the MISP modules that provide extensive capabilities to the system, and
- the information flow of threat intelligence towards to end- users.

2.2 Relations to other activities in the project

This deliverable (5.2) derives from T5.1 "Threat intelligence techniques" and adopts the knowledge gained from T2.4 "Threat sharing and awareness" and the D2.5 "Threats actors' attack strategies", focused on Cyber-Trust framework. Particularly, this deliverable (D5.2 "Cyber-Threat Intelligence Sharing") which is due on M21 gives input to D5.5 "Cyber-threat intelligence: architecture and methods", which is due on M30. More specifically, this input will provide information that will assist to the architectural aspects of the eVDB. D5.5 will describe the architecture of the cyber-threat intelligence gathering tool, the methods/algorithms explored and developed, as well as research results obtained from the experimental setups.

2.3 Structure of the document

This document is comprised of the following six (6) sections:

- Section 1 abstracts the deliverable (D5.2).
- Section 2 gives information regarding D5.2 content and correlated activities.
- Section 3 describes the cyber-threat intelligence sharing technologies in MISP.
- Section 4 emphasises on the way that the CTI will be flow to end-users and devices within the Cyber-Trust project.
- Section 5 describes how the cyber-threat intelligence is going to be viewed by end-users through User Interface (UI).
- Section 6 describes the legal framework of the project regarding the sharing of information.
- Section 7 concludes the deliverable.

3. Threat intelligence sharing services (Enriched Vulnerability Database (eVDB) - A07+A09)

In general, Threat Intelligence Sharing Services (eVDB) is an aggregated unique technological solution consisted by a combination of two parts, eVDB Admin [A07] and Sharing Service [A09]. Enriched Vulnerability Database (eVDB) is a component of the Cyber-Trust (CT) platform. Specifically, it is a database which provides enhanced and scalable cyber-threat intelligence, which will enhance Cyber-Trust functionalities especially integrity storage, threat proactiveness and sharing capabilities. Each of two parts has its own dedicated capabilities and responsibilities which are combined in cases.

Initially, eVDB Admin is responsible for:

a) preserving information received from components of the CT platform and especially from the CT crawler and
b) disseminating the data through the Sharing Service and feed other CT components such as Trust Management Service (TMS) and Intrusion Detection Service (IDS) which will be able to utilize the respective information to detect threats.

From the beginning of the project, in D2.2 “Threat sharing methods: comparative analysis”, a survey was conducted among a variety of existing mechanisms and platforms, for choosing the sharing mechanism and sharing platform of the Cyber-Trust. Finally, through the comparative analysis, the sharing mechanism that was selected is Structured Threat Information Expression (STIX) and the sharing platform is Malware Information Sharing Platform (MISP).

3.1 Overview/objectives

The Cyber-Threat Intelligence Services aim to:

- Create a secure vulnerability database.
- Achieve common situational awareness across organizations.
- Create a scalable database which will provide:
  - Sharing of information between specific Cyber-Trust components.
  - Sharing of capabilities.
- Provide a simple, yet flexible, collaborative way of characterizing and categorizing threat activity that supports analysis, senior level decision making, and cybersecurity proactive system.
- Facilitate cyber threat trend and gap analysis, assessment of collection posture.

3.2 Functional coverage

The functional coverage of eVDB encompasses the functional and non-functional requirements of the component. The eVDB requirements were gathered from the end-user questionnaires, the state-of-the-art analysis of relative technological advanced tools. Also, some requirements were derived from the operational needs of the eVDB in order to be connected with other apps, protocols, tools, etc. Finally, the functional and non-functional requirements of the eVDB fed the architectural requirements and structured some architectural specifications of the platform. The architectural requirements will not be included in this deliverable since they consist restricted information.

3.2.1 Related requirements

In Table 1 we present the functional requirements of the CT platform related to eVDB.

Table 1: Functional requirements related to eVDB

ID. FR Requirement	Definition
FR57	The user will be able to select one or more of his/hers registered devices (through the Web portal) and through the eVDB search tool will search for vulnerabilities regarding the selected devices.

FR63	Users will be able to search and retrieve information regarding security issues and intelligence that pertain to their devices (see NFR25)
FR76	The user (e.g. Security officer) will be able to create the cyber-attack graphical security model based on specific network infrastructures (architecture, topology, devices and related information).
FR77	Development of appropriate UI for entering dynamic parameters regarding the system (i.e. state transition model, expected utility function). These parameters will be used in order to re-calculate attack's likelihood and success probability.

In Table 2 we present the non-functional requirements of the CT platform related to eVDB.

Table 2: Non-functional requirements related to eVDB

ID. NFR Requirement	Definition
NFR3	Strict access rights
NFR18	Open Source Threat Intelligence Platform (MISP) will be used and extended as necessary in order to be used for sharing the respective information
NFR20	Creation of the Enriched Vulnerability Database (eVDB)
NFR23	Development of eVDB search and discovery tool.
NFR24	Development of appropriate query interface based on the access role of the user (to retrieve info from eVDB)
NFR25	The platform must have "Review and curate vulnerabilities" functionality

3.2.2 Related use cases

The requirements presented in Section 3.2.1 relate to the Cyber-Trust use cases that are presented in Table 3 and Table 4.

Table 3: Use case relating to the functional requirements of eVDB

ID. FR Requirement	Cyber-Trust use cases related to the requirements
FR57	UCG-02-05: Register to the eVDB sharing service
	UCG-05-08: Visualize known and zero-day vulnerabilities
	UCG-06-04: Query and retrieve information from eVDB
	UCG-14-08: Match device profile with eVDB content
FR63	UCG-02-05: Register to the eVDB sharing service
	UCG-05-08: Visualize known and zero-day vulnerabilities
	UCG-06-02: Raise alert for device owner
	UCG-06-04: Query and retrieve information from eVDB
	UCG-06-05: Review and validate eVDB entries
	UCG-14-07: Notify about updates and security-related issues
	UCG-14-08: Match device profile with eVDB content
	UCG-16-05: Crawl the clear/deep/dark web and update the eVDB
FR76	UCG-05-08: Visualize known and zero-day vulnerabilities
	UCG-06-04: Query and retrieve information from eVDB
	UCG-14-08: Match device profile with eVDB content

FR77	UCG-14-08: Match device profile with eVDB content

Table 4: Use case relating to the non-functional requirements of eVDB

ID. NFR Requirement	Cyber-Trust use cases related to the requirements
NFR3	UCG-06-06: Provide feedback/rating on sources of vulnerabilities
NFR24	UCG-06-04: Query and retrieve information from eVDB
NFR25	UCG-06-05: Review and validate eVDB entries

3.3 Technology update

3.3.1 Why we choose MISP

In D2.2 we have illustrated that CTI sharing provides great benefits, but also has to deal with challenges, such as establishing trust, achieving interoperability and automation, securing sensitive information and enabling information sharing. From these challenges, a set of requirements is thoroughly inferred. Using such requirements, we compared and evaluated several CTI sharing platforms. The requirements that need to be met by the sharing platform of choice, first, demand from the selected platform to allow CTI sharing between the platform and different stakeholders, along with the end-user's devices. Next, the sharing mechanism and platform should be expressible, flexible, scalable, and open source. Moreover, it should allow storing information about the source of CTI. Furthermore, it should facilitate automation and provide CTI in both human and machine-readable formats. Finally, it should support information filtering and alerting functionalities. Regarding these aspects, we concluded that Malware Information Sharing Platform (MISP), is the most suitable platform to act as the project's eVDB.

MISP is an *open source threat intelligence and open standard for threat information sharing platform*, which is able to store and share technical and non-technical information about malware samples, incidents, attackers and intelligence. Specifically, MISP provides a user interface (UI), which enables users to create, search or share events amongst other MISP users or communities. Furthermore, all CTI stored in the MISP database can be accessed through an API, which allows for data exporting in a wide variety of formats, such as XML, JSON, OpenIOC, STIX, and more.

Additionally, MISP has an automatic correlation mechanism that is able to identify relationships between attributes, objects and indicators from malware correlation engines. Moreover, MISP stores data in a structured format, provides extensive support of cyber-security indicators for different vertical sectors, and supports CTI sharing for both human and machine applications. More details about MISP functionalities are described in https://github.com/MISP/MISP.

Intelligence vocabularies (MISP galaxy) can be bundled with existing threat adversaries, malware and ransomware or linked to events from MITRE ATT&CK, which is a publicly available knowledge base, that contains adversary tactics and techniques based on real observations. Communities can leverage MITRE ATT&CK, in order to develop specific threat models and methodologies for Tactics, Techniques and Procedures (TTPs).

Finally, MISP provides a flexible free text import tool to facilitate the integration of unstructured reports into MISP and an adjustable taxonomy to classify and tag events according to the users' own classification schemes and taxonomies.

3.3.2 Current state of MISP

MISP is an active open-source platform, which is enhanced, fixed, and introduced with additional support, approximately on a monthly basis. Currently, we use the latest version of MISP (2.4.119), which has been released on December 2, 2019.

Below, we provide a brief summary of notable changes that fulfil the project's expectations and illustrate MISP capabilities, as they were extracted from its release page:

MISP 2.4.95 (2018-09-06)

- The search API in MISP has been refactored to be consistent among the various export formats (JSON, XML, OpenIOC, Suricata, Snort, and the text export); particularly, regarding the filtering process. String searches are by default exact lookups, but the search API allows the use of "%" wildcards to perform substring searches.
- A complete REST client has been added in the MISP interface, to enable MISP users query the API from the instance at hand.
- A debug functionality has been added in any API query to quickly show the SQL queries performed.

MISP 2.4.96 (2018-10-09)

- All MISP export APIs have been unified into the restSearch APIs, with an improved query format.
- A pagination system has been introduced, allowing users to easily paginate over search result sets and limit the output.
- The search results in the MISP UI can be directly downloaded in any of the supported formats available in MISP.
- Event/attribute data fetching performance increased, with the use of an internal pagination and caching mechanism, which scales with the amount of memory given to the PHP process, and hence reducing the chance of running into memory limit issues.
- The freetext import is now delegated to a background process for large imports.

MISP 2.4.98 (2018-11-26)

- Improved UI consistency (e.g. attributes search output).
- Improved error handling and error messages.

MISP 2.4.100 (2018-12-31)

- Improvements to the UI, API, import and export.
- Addition of a new query builder, available through the REST client interface, that facilitates users to create JSON queries.

MISP 2.4.101 (2019-01-20)

- Improvements to the UI, import and export.
- Enabling/Disabling correlations is now accessible when creating/modifying an attribute.

MISP 2.4.103 (2019-03-04)

- Improvements to the UI.
- Implementation of a new attribute filtering tool to the event view, that allows for complex filtering rules.

MISP 2.4.106 (2019-04-25)

- Performance improvements for events with large numbers of attributes and objects.

MISP 2.4.108 (2019-06-04)

- Added object_relation as a filter for both the event/attribute restSearch functions.

MISP 2.4.109 (2019-06-13)

- Added date as a new restSearch filter, with a variety of accepted syntax options, such as:

- time ranges in the shorthand format (7d or 24h, etc.)
- Timestamps
- fallback parsing for other formats (2019-01-01, "fortnight ago", etc.)
- date ranges using lists [14d, 7d]

MISP 2.4.112 (2019-08-02)

- New parameters added to attributes/restSearch to include additional context.
  - includeCorrelations: includes the correlations to other attributes (includes a light-weight event object with each attribute)
  - includeContext: includes the additional event fields in the attributes/restSearch results (in JSON format) (e.g. UUID)
- Added "weakness" object. It describes a weakness through the Common Weakness Enumeration (CWE) format.

MISP 2.4.114 (2019-08-30)

- Added a new diagnostic tool, which allows administrators to keep track of the database table sizes in MISP, along with the potentially recoverable space by optimizing the table.

MISP 2.4.117 (2019-10-10)

- Added user settings. All configuration options in MISP have been based on system-wide, organization-wide or role-based configurations. The new user settings system allows for the configuration on the user level.
- Performance improvements both on MISP and PyMISP, regarding events that include large amounts of objects and attributes.
- Introduction of a new set of options for administrators to enforce requests rate limits on API users.

MISP 2.4.118 (2019-11-08)

- Improved the database schema model update module in MISP. That enables administrators view the current inconsistencies of any past model change or the ongoing upgrade of the database model.

MISP 2.4.119 (2019-12-02)

- Enhanced database diagnostics with the integration of a new sub-system that compares the current state of the MISP database to the reference DB schema, highlighting potential issues or divergences. Additionally, it allows users to generate SQL queries that would rectify the potential issues.
- Improved timestamp filtering in MISP. It now provides 4 different timestamp filters on the following levels: event, attribute, attribute and event, and event publish.
- Added tracking of the API deprecations, warning users of their state.

Generally, it is strongly suggested to keep MISP up to date in accordance with the latest version published, in order to fully exploit the platform's improvements and fixes.

3.3.3 Technology stack and applied tools

As mentioned in Section 3.3.2, we currently work with MISP v2.4.119, which is the latest version of MISP published. MISP is built upon programming frameworks like CakePHP and PHP for the UI, and MariaDB/MySQL for the data storage.

In this project, we use the technologies of Table 5 to support the implementation of the eVDB with MISP v2.4.119.

Table 5: Technologies support the eVDB implementation

Technologies	Description
CakePHP 2.10.19	CakePHP is an open-source web framework, which follows the Model-View-Controller (MVC) approach and is written in PHP. MISP is built upon CakePHP v2.10.19, which supports PHP v7.0+, and it also makes use of the CakeResque plugin of CakePHP, which enables the creation of background jobs that can be processed offline.
PHP 7.2	We make use of the PHP 7.2 version for the implementation of the eVDB, upon which CakePHP builds the MISP platform, which also connects the web application with the data storage.
MariaDB 10.1	MariaDB is a variation of the MySQL RDBMS. MariaDB acts as the eVDB data storage, where MISP stores all required web app data structures, along with all CTI that is gathered and can be queried.

Additionally, we use of the technologies illustrated in Table 6 to support the gathering and distribution of CTI.

Table 6: Technologies support gathering and distribution of CTI

Technologies	Description
PyMISP 2.4.119	PyMISP is a Python programming language library that provides access to the MISP platform via its REST API. It enables users to fetch events, add, update, delete and search events/attributes or samples. Through the utilization of PyMISP library, by providing all required data for the authorization of the user registered in MISP, we create/update events, each time we gather new CTI from our monitored sources. Furthermore, it facilitates the creation of scripts that enable other components to easily interact with MISP. PyMISP 2.4.119 is supported by Python 3.6+ versions.
Python 3.6	We use Python 3.6 to automate the process of collecting CTI from our monitored sources. Through the implemented python scripts, we gather CTI from NVD, JVN, VulDB, KB-Cert and Exploit-DB. All gathered CTI is then structured into JSON objects that can be interpreted as MISP objects through the PyMISP library and finally inserted into the eVDB.
ZeroMQ/misp-dashboard	We use ZeroMQ, which is integrated into the MISP platform, and allows for the implementation of the publish/subscribe functionalities, that the component provides. ZeroMQ is a topic-based publish/subscribe mechanism that enables the distribution of CTI in channels that filter it by events, attributes, user, organization and their combinations.

3.3.4 Example of PyMISP usage

In this subsection, we will provide an example of a python script that given the event ID as it is registered in MISP, it will return specific information about this event from the MISP vulnerability objects that exist within it. This is achieved through the utilization of PyMISP library, that enables the communication with the MISP REST API.

An example of PyMISP library application is presented in the script of Figure 1. This script presents the information that is stored in MISP about a specific event.

```
from pymisp
import ExpandedPyMISP
from keys
```

```
import misp_url, misp_key, misp_verifycert
from datetime
import datetime
import argparse
import os
import json
if __name__ == '__main__':
    parser = argparse.ArgumentParser(description = 'Get all the events matching a value for a given param.')
parser.add_argument("-s", "--search", required = False, action = 'store_true', help = "Search flag for searching information regarding a specific CVE ID.")
parser.add_argument("-d", "--id", required = False, help = "Limit results per page.")
args = parser.parse_args()
misp = ExpandedPyMISP(misp_url, misp_key)
if args.search is not False:
    rel_events = []
rel_events_ids = []
timestamp = 1545730073
dt_object = datetime.fromtimestamp(timestamp)
cve_id = ''
vuln_confs = []
pubtime = ''
description = ''
modtime = ''
cvss_score = 0.0
cvss_str = ''
refs = []
summary = ''
credit = ''
if args.id is not None:
    result = misp.search(eventid = args.id)
if not result:
    print('No results.')
exit(0)
else :
    for r in reversed(result):
    rel_events = r['Event']['RelatedEvent']
for r2 in rel_events:
    rel_events_ids.append(r2['Event']['id'])
rel_events_ids = reversed(rel_events_ids)
dt_object = datetime.fromtimestamp(int(r['Event']['Object'][0]['timestamp']))
for att in r['Event']['Object'][0]['Attribute']:
    if att['object_relation'] == 'id':
    cve_id = att['value']
if att['object_relation'] == 'vulnerable_configuration':
    vuln_confs.append(att['value'])
```

```
if att['object_relation'] == 'published':
    pubtime = att['value']
if att['object_relation'] == 'description':
    description = att['value']
if att['object_relation'] == 'modified':
    modtime = att['value']
if att['object_relation'] == 'cvss-score':
    cvss_score = att['value']
if att['object_relation'] == 'cvss-string':
    cvss_str = att['value']
if att['object_relation'] == 'references':
    refs.append(att['value'])
if att['object_relation'] == 'summary':
    summary = att['value']
if att['object_relation'] == 'credit':
    credit = att['value']
print("CVE:", cve_id)
print("Event datetime:", dt_object)
print("Related Events IDs:")
for r in rel_events_ids:
    print("\t", r)
print("Summary:", summary)
print("Publication datetime:", pubtime)
print("Last modification datetime:", modtime)
print("CVSS string:", cvss_str)
print("CVSS score:", cvss_score)
print("Vulnerable Configurations:")
for vc in vuln_confs:
    print("\t", vc)
print("References:")
for r in refs:
    print("\t", r)
print("Credit/Source:", credit)
print("")
print("")
print("Description:")
print(description)
else :
    print("Usage: python3 search.py -s -d <event ID>")
print("Please define the event ID of interest.")
```

Figure 1: Using PyMISP to access information stored in MISP

If the script of Figure 1 is stored in a file named `example.py` it can be executed using `python3 example.py -s -d <event ID>`, where `<event ID>` is the ID of the event we are interested in. A sample execution is presented below.

```
$ python3 example.py -s -d 83255

CVE: CVE-2009-5154
Event datetime: 2019-09-07 20:37:11
Related Events IDs:
     4020
     4019
     4018
Summary: An issue was discovered on MOBOTIX S14 MX-V4.2.1.61 devices. There
is a default password of meinsm for the admin account.
Publication datetime: 2019-02-09T17:29:00.247-05:00
Last modification datetime: 2019-02-13T11:10:06.227-05:00
CVSS string: CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H
CVSS score: 9.8
Vulnerable Configurations:
     cpe:/o:mobotix:s14_firmware:mx-v4.2.1.61
     cpe:/h:mobotix:s14:-
```

More examples are can be found in [1] [2]

PyMISP facilitates the communication between various components and MISP, and the automation of processes that require the querying of the eVDB. Similarly, we extract information from our monitored sources and insert them into the eVDB, in a structured manner.

3.3.5 The MISP eVDB deployment on OTE

Currently, we have deployed a dockerized version of the project's eVDB on OTE infrastructures, which can only be accessed through a private protected network and monitors the defined sources daily.

The eVDB is split in two docker containers which communicate with each other through the ports that are defined in the docker-composer file. One docker container is for the MISP web UI and the other is for the data storage of MISP, which contains a MariaDB instance. Next, through python scripts that use the PyMISP library to communicate with the MISP REST API, which exists in the first container, we can create or modify MISP Events, to include all recently gathered CTI that concern specific CVE IDs. Finally, through a crontab, the python scripts run daily, in order to keep our eVDB entries up to date.

3.4 Application architecture

MISPs' core functionality is sharing. Everyone may have interchangeably the role of the consumer or the producer. In general, MISP sharing capability gives the opportunity to the system to take the already known information and form it to enhanced information based on the contributed attributes which are being taken from various sources. The new event with the contributed attributes will be stored in the central database of MISP and will be available to all users (see Figure 2).

Cyber-Trust MISP will gather the targeted and malicious information and will transfer it into Cyber-Trust components. The information of eVDB repository will feed MISP as well.

[1] https://gist.github.com/llandeilocymro/7dbe3daaab6d058d609fd9a0b24301cb

[2] https://www.use-ip.co.uk/forum/threads/mobotix-default-password.76/.

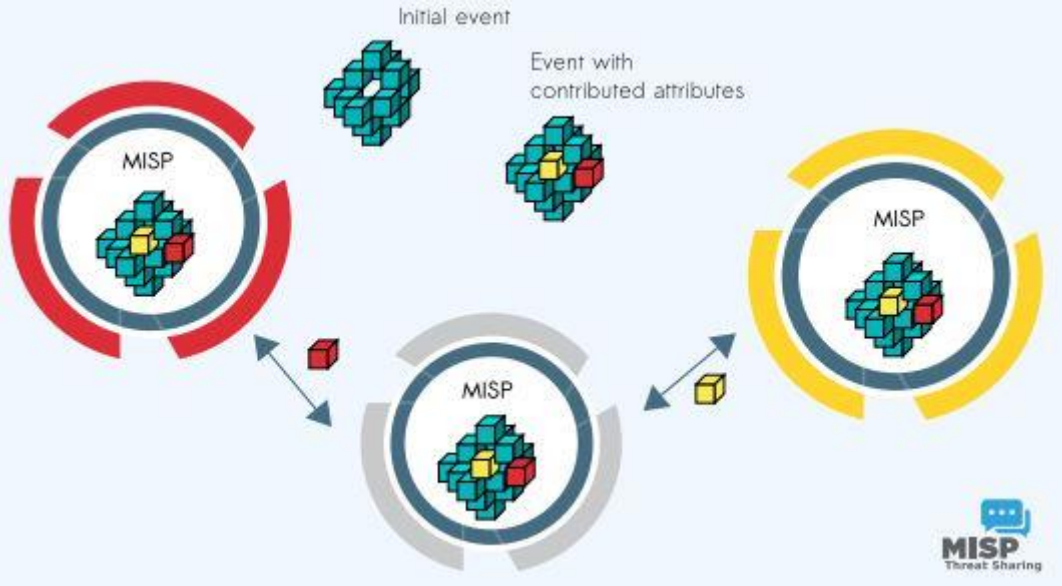

Figure 2: High level view of MISP's sharing capability [5]

MISP [6] contains various data categories which belonged to different incident families. Namely, the incident families are MISP events, attributes, objects, indicators etc. Below, we provide the definitions of MISP's terminologies.

MISP Events

- Events are encapsulations for contextually linked information.

MISP Attributes

- Attribute is any information that characterizes malicious intention.

MISP Galaxies

- Each MISP Galaxy [7] is a method to express a large object that can be attached to MISP events or attributes. These methods endeavour to contextualize, classify and classified data based on threat actors, preventive measures, tools used by adversaries.

MISP Objects

- MISP objects are added to MISP modelling to extend and advanced the combinations of attributes. Attribute compositions describing points of data using many facets, constructed along the lines of community and user defined templates. The creation of these objects and their associated attributes are based on real cyber security use-cases and existing practices in information sharing. The objects are just shared like any other attributes in MISP even if the other MISP instances do not have the template of the object.

MISP clusters

- A cluster is a large object which is composed by of one or more elements. Elements are expressed as key-values. In MISP galaxy there are standard vocabularies (default vocabularies), but users have the capability to modify and update them. Vocabularies are from existing standards (like STIX, Veris, MISP and so on) or are customizable.

MISP Indicators

- Indicators are patterns that can be used to detect suspicious or malicious cyber activity.

MISP Attackers' techniques

- MISP integrates at event or attribute level MITRE's Adversarial Tactics, Techniques and Common Knowledge (ATT&CK) [11]. Data and events should not be viewed in isolation but as part of a chain of behaviour that could lead to other activities based on the information obtained. MITRE ATT&CK translates technical data or IoCs into cyber-threat intelligence and visualize it through Maltego tool.

Indicator of Compromise (IoC)

- IoC is an artifact observed on a network or in an operating system or information channel that could reference an intrusion or a reference to a technique used by an attacker. IoCs are a subset or indicators. Indicators contain a pattern that can be used to detect suspicious or malicious cyber activity.

Table 7: MISP incident families correspond to the related data

MISP incident families	Related Data Categories
MISP Events	Different groups of information categories
MISP Attributes	Event packages, vulnerabilities, malicious information, network indicators (e.g. malicious IP Address), system indicators (e.g. string in memory), malicious bank account details, etc.
MISP Objects	There are a variety of *object templates* [8], such as, **tsk-chats** (an object template to gather information from evidential or interesting exchange of messages identified during a digital forensic investigation. Attributes: message type, date-time sent/receive, source, destination, app-used, subject, message, attachments, additional comments) **tsk-web bookmark** (an object template to add evidential bookmarks identified during a digital forensic investigation. Attributes: URL, datetime bookmarked, name, title, browser, domain-name, domain-ip, additional-comments) etc.
MISP Indicators	IoC (Indicator of Compromise) (e.g. hashes etc.) is a subset of indicators/Network indicators/system indicators etc.
MISP Galaxies	Cyber-threat actors, preventive measures, malicious cyber-tools.
MISP attackers' techniques	ATT&CK data. In the ATT&CK knowledge base are included threat models and methodologies that reveal tactics patterns in private and governmental sector as well as in cyber security products and service community.
Indicator of Compromise (IoC)	IoC could be hashes, malicious IP address, URLs, email address, etc.

In Table 7 we present the data that appear in each incident family.

3.4.1 General MISP layout

The MISP layout differentiates whether the end-user is a simple user or the administrator of the platform [9].

3.4.1.1 Simple user

The *top bar* of a simple user's interface (see Figure 3) includes the tabs described below:

- **Home** tab guide the user to the initial profiling interface of the application.
- **Event actions** gives access to all users to functionalities that are related to creation, modification, deletion, publishing, searching and listing of the events and attributes.
- **Galaxies** guide the user to the list of MISP Galaxies on the MISP instance.
- **Input filters** define the type of data that enter in each instance. The tab "Input filters" has a drop-down list with various options. "Import Regexp" allows the admin of the system to view the Regular Expression rules which define the data that entered into the system, a user with regex
- **Global Actions** allows the user to have access to information regarding MISP and a specific instance, also has the capability to view and modify the profile, receive a manual of MISP. Some options include information regarding the latest MISP news, the sharing groups that the organisation communicate, organisation role permissions etc. Also, administrator can view and manage profiling details, can view organisations that exists on a specific instance as well as the statics which are referred to the users and the data on this instance.
- **MISP** tab provides a link that leads to the baseURL.
- **User** (in Figure 3: Simple user's Figure 3 is the "Steve" tab) is auto generated from the user email address of current logged in user.
- **The envelop icon** guides the user to the User Dashboard, which contains the latest information of the account's management such as, notifications, modifications of the account etc.
- **Log out** leads you out of the system.

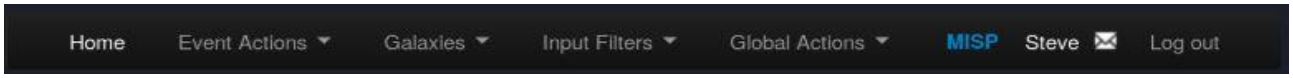

Figure 3: Simple user's top bar

3.4.1.2 Administrator

The *top bar* of an administrator's Interface (see Figure 4) includes the tabs are described below.

- **Home** tab guide the administrator to the initial profiling interface of the application.
- **Event actions** gives access to all functionalities related to creation, modification, deletion, publishing, searching and listing of the events and attributes.
- **Galaxies** guide the administrator to the list of MISP Galaxies and enables him to update the galaxy as well.
- **Input filters** has a drop-down list with various options. "Import Regexp" allows the admin of the system to view the Regular Expression rules which define the data that are inserted into the system. Therefore, a site administrator or a user with regex permissions can edit the rules. "Signature Whitelist" includes the kind of information that should be forbidden by the system, and the site administrator can edit this list. "List warninglists" includes indicators for potential false, positives, errors or mistakes. The warning lists are integrated in MISP to display an info/warning box at the event and attribute level.
- **Global Actions** allows the user to have access to information regarding MISP and a specific instance, also has the capability to view and modify the profile, receive a manual of MISP. Some options include information regarding the latest MISP news, the sharing groups that the organisation communicate, organisation role permissions etc. Also, the administrator can view and manage profiling details, can view organisations that exist on a specific instance as well as the statistics, which are referred to the users and the data on this instance.
- **Sync Actions** prerequires administrator's access rights, then the admin can visualize the instances connections. Sync Actions includes "List Servers" and "List Feeds".
- **Audit** needs permission to be accessible. The administrator can visualize organisation logs (or for site admins for the entire system) and search targeted the logs of a specific event.
- **MISP tab provides** a link that leads to the baseURL.

- **Admin** can handle user's information. More specifically, view, modify, delete and add users in the systems. For coordination issues or in case of any problem in user's accounts, the admin has the capability to contact the current and future users and provide them temporary passwords. The admin has the same capabilities as before, towards the organisations.
- **The envelop icon** guides the user to the User Dashboard, which contains the latest information of the account management such as, notifications, modifications of the account etc.
- **Log out** leads you out of the system.

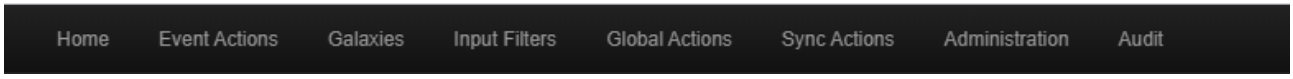

Figure 4: Administrator's top bar

Finally, there is a *left bar* that changes based on each page-group. The blue selection shows the number of the page that you are on a specific time.

3.4.2 Events

As it was previously referred "Event actions" gives access to all users, to functionalities that are related to creation, modification, deletion, publishing, searching and listing of the events and attributes. Some of the aforementioned functionalities are presented below.

Figure 5: Layout in the List of Events

3.4.2.1 Creating an event

In order to create an event, you need to make three (3) actions [6].

- Generation of the event itself. This means that the basic event will be created without any actual attributes and will store general information, such as description, time and risk level of the incident.
- Populating the event with attributes and attachments by clicking on the tab "New Event" and completing the particular form.
- Publishing the event.

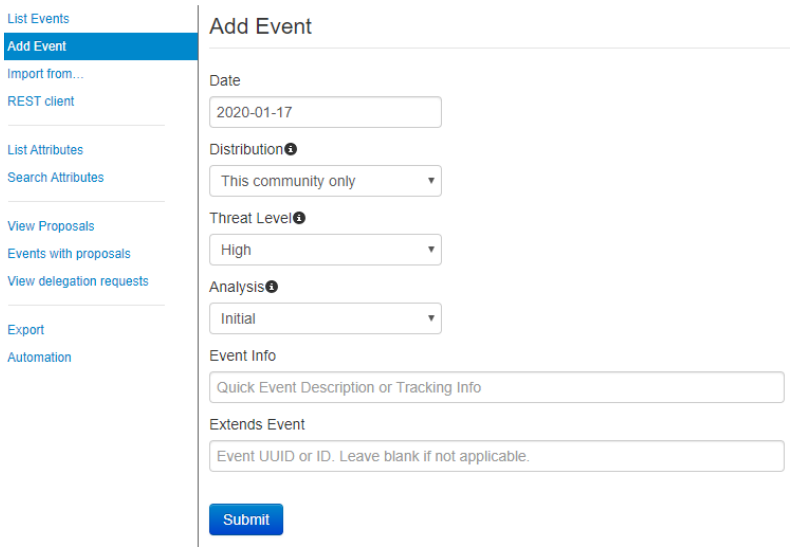

Figure 6: Adding process of an Event in MISP

Every user will complete the fields with the exact information. The user should pay attention through the data completion since they are consisting vital elements of the incident's description.

- **Date** indicates the date of the incident
- **Distribution** is a setting control, that reveals who can see the event, once it becomes published and eventually when it is pulled. Also, you can control whether the event will be shared to other servers too or not.
- **Threat level** indicates the risk level of an event. Incidents can be classified into three (3) threat categories, more specifically a) low, b) medium, c) high. Also, this field can remain unclarified.
- **Analysis** specifies the event's stage of the analysis, more specifically a) Initial, b) ongoing, c) completed.
- **Event Description** gives information regarding malware/incident with a brief description starting with the internal reference. The system replaces the detected text strings that are in accordance with the administrator's regular reference expression.
- **GFI Sandbox** gives the capability to upload the exported documents of the aforementioned malware analysis tool.

3.4.2.2 List of events

Here you will find information regarding the interface of MISP that allows the user to view, search for events and attributes of events that are already stored in the system in various ways. The menu, through the tab "List events", allows for the creation of a list with the 60 last events in the system, without presenting the attributes [6].

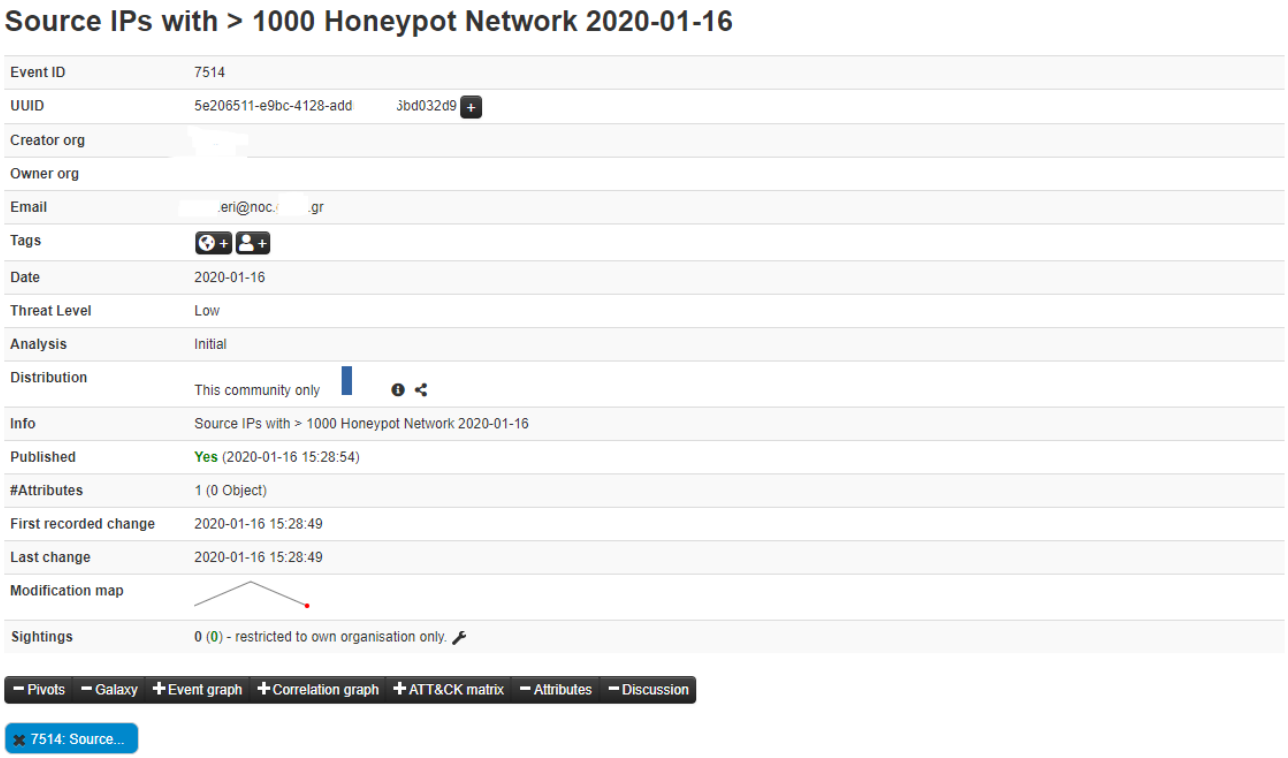

Figure 7: View of an Event in the MISP

The event in the MISP platform is a tab that encompasses a filter mechanism. Specifically, these filters are indicated below:

- **ID** shows the ID of the event.
- **Uuid** provided in order to avoid collisions between events and attributes (during for example a sync) a Uuid is assigned that uniquely identifies each of them.
- **Org** is referred to the organisation that has originally created the event. The logo (if it exists on the server, alternatively a string) representing the organisation is also shown int he right upper corner.
- **Contributors** shows a list of the organisations that have contributed to the event via proposals. If a user clicks any of the logos listed here, MISP will redirect to a filtered event history view, including only the changes made by the organisation.
- **Tags** shows a list of tags associated with the event. Clicking a tag will show a list of events with the same tag attached. The little cross next to each tag allows the users to remove the tag from the event, whilst the '+' button allows them to assign a tag. For the latter two options to be visible, the users should have tagging permission.
- **Date** indicates the date of detection, set by the user that creates the event, not to be confused with the creation date of the event.
- **Threat Level** indicates the assigned threat level of the event.
- **Analysis** provides the status of the analysis.
- **Distribution** shows the distribution rules applied to this event, controlling whether only the authoring organisation can see (Your organisation only) it or everyone on the instance (This community only). The two remaining settings allow the event to be propagated to organisations on remote connected instances.
- **Info** tab gives a short description of the event itself. Make sure not to put information in here that could be used for correlation purposes and be better suited as an Attribute.
- **Published** gives information whether the event has been published or not. Publishing allows the attributes of the event to be used for all eligible exports and it notifies users that have subscribed to the event alerts. Also, a publish initiates a push to all eligible instances.

Also, there is the *List of Related Events*. This list is referred to the relations that are shown on the right-hand side of the general event information. Events can be related by having one or more attributes that are exact matches. For example, if two events both contain a source IP attribute of 11.11.11.11 then they are related. The list of events that are related the currently shown one, are listed under "Related Events", as links (titled the related event's date and ID number) to the events themselves.

3.4.3 eVDB storage and sources

To store the CTI gathered from our monitored sources, we have followed an approach through which we encapsulate all related MISP objects, along with their corresponding attributes, into one MISP event for each CVE ID. The sources that we periodically monitor are NVD, JVN, VulDB, KB-Cert and Exploit-DB. Below, we present a table of all predefined MISP objects, along with their attributes, that are used for storing that information, as they were extracted by the MISP documentation:

Table 8: Predefined MISP objects with their attributes

MISP Objects	Attributes	Attribute Type	Description
vulnerability	id	text	Vulnerability ID (generally CVE, but not necessarily). The id is not required as the object itself has an UUID and the CVE id can be update or assigned later.
	description	text	Description of the vulnerability
	summary	text	Summary of the vulnerability
	vulnerable_configuration	text	The vulnerable configuration is described in CPE format
	modified	datetime	Last modification date
	published	datetime	Initial publication date
	references	link	External references
	cvss-score	float	Score of the Common Vulnerability Scoring System (version 3)
	cvss-string	text	String of the Common Vulnerability Scoring System (version 3)
	credit	text	Who reported/found the vulnerability such as an organisation, person or nickname
weakness	id	text	Weakness ID (generally CWE)
	description	text	Description of the weakness
	name	text	Name of the weakness
exploit-poc	description	text	Description of the exploit - proof of concept
	vulnerable_configuration	text	The vulnerable configuration described in CPE format where the exploit/proof of concept is valid
	author	text	Author of the exploit - proof of concept
	references	link	External references

Additionally, in Table 9Table 9: A Custom MISP object we present a custom MISP object, which we created for the purposes of storing all available information gathered by VulDB:

Table 9: A Custom MISP object

MISP Objects	Attributes	Attribute Type	Description
vuldb-vulnerability	id	text	Vulnerability ID (generally CVE, but not necessarily). The id is not required as the object itself has an UUID and the CVE id can be update or assigned later.
	description	text	Description of the vulnerability
	summary	text	Summary of the vulnerability
	published	datetime	Initial publication date
	cvss-score	float	Score of the Common Vulnerability Scoring System (version 3). This is a Meta score, calculated by vuldb. The calculation method will be described in comment. [e.g. CVSSof(vuldb+nvd)/2]
	status	text	Status of the vulnerability approval
	cvss-string-VDB	text	String of the Common Vulnerability Scoring System (version 3) of vuldb security analysts
	cvss-string-NVD	text	String of the Common Vulnerability Scoring System (version 3) of nvd security analysts
	cvss-string-Vend	text	String of the Common Vulnerability Scoring System (version 3) of vendor security analysts
	cvss-string-Res	text	String of the Common Vulnerability Scoring System (version 3) of researcher who analyzed it for vuldb
	cvss-tmp-score	float	Score of the Temporal Common Vulnerability Scoring System (version 3). This is a Meta score, calculated by vuldb, as an average score of different sources, to provide a normalized scoring system
	cti-interest-score	float	Vuldb CTI team is monitoring different web sites, mailing lists, exploit markets and social media networks. The CTI Interest Score identifies the interest of attackers and the security community for this specific vulnerability in real-time. A high score indicates an elevated risk to be targeted for this vulnerability
	vuldb-link	link	The link to the vuldb advisory
	zeroday-price	text	Vuldb analysts are monitoring exploit markets and are in contact with vulnerability brokers. The range indicates the observed or calculated exploit price to be seen on exploit markets. A good indicator to understand the monetary effort required for and the popularity of an attack. This is the price range of the exploit for the 0day exploitation of the vulnerability
	current-price	text	Vuldb analysts are monitoring exploit markets and are in contact with vulnerability brokers. The range indicates the observed or calculated exploit price to be seen on exploit markets. A good indicator to understand the monetary

			effort required for and the popularity of an attack. This is the price range of the exploit for the exploitation of the vulnerability in a specific moment. (The date will be provided in a comment)
	exploitability	text	The likeliness of an exploit of the vulnerability to happen
	remediation	text	A status of whether there is a remediation for this vulnerability
	credit	text	Who reported/found the vulnerability such as an organisation, person or nickname

Finally, in Table 10, we present the structure of a complete CVE event's objects, that represents all information gathered from the monitored sources.

Table 10: Structure of a complete CVE event's object

Source	MISP Objects
NVD	vulnerability
JVN	vulnerability
VulDB	vuldb-vulnerability
KB-Cert	vulnerability
Exploit-DB	exploit-poc

3.4.4 Correlation engine

Correlation engine encompasses all the correlations between attributes and more advanced correlations like Fuzzy hashing correlation (e.g., ssdeep) or CIDR block matching. Correlation can be also enabled, or event disabled per attribute.

The value or value-pair of the attribute is the main payload of the attribute, which is described by the category and type columns. For certain types of attributes that are made up of value-pairs the two parts will be split by a pipe (|), such as for filename|md5. The value field(s) are used by the correlation engine to find relations between events. In value-pair attributes both values are correlated individually.

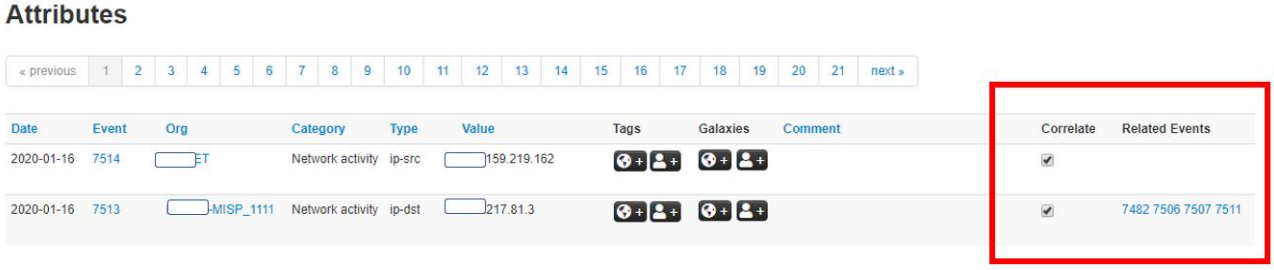

Figure 8: Correlation Engine of MISP

Currently, the way correlation information is stored is very space demanding. To illustrate an example, in a setting that stores 83 thousand events that have 5 million attributes, the correlation information consists of 1.2 billion records that occupy over 315 Gb of hard disk storage. In our operational mode, we expect more than 135 thousand events. Thus, our estimation of the size of correlation information will exceed 3 billion

records and 1 Tb of disk storage. Note also that this information will constantly increase. For instance, during 2019, over 17 thousand new events where added. It is very likely to expect that the number of new events will constantly increase.

In total, the current storage of correlation information should be revisited. Otherwise, even if modern disk storages could fit this increasing volume of information, it will surpass the processing abilities of the database management system. Thus, we are currently working on finding solutions to decrease this vast storage requirement.

3.5 Application Programming Interfaces (APIs)

MISP encompasses a variety of modules which assist to the cyber-threat information collection, exchange, correlation, importing, exporting etc. It also provides a variety of existed tools for the handling and modification of new information and the utilization and storage of existed knowledge. Having as a short-term goal, the cyber-threat information gathering and analysis, and as a long-term goal, the cyber-threat prediction and security enhancement.

MISP modules and software applications are autonomous tools that can be used for expanding the already existing capabilities and services. The software is written in Python 3 following a simple API interface. The goal of MISP APIs is to modify and extend the capabilities of software without changing the components.

These software applications and tools are connected with the system through modules. There are three (3) different categories of modules a) expansion modules, b) export modules, c) import modules. In addition to the modules, there are software/services that are supported by the MISP platform and extend its functionalities. Expansion modules are divided into two types, expansion type and hover type; a part of the expansion modules are set by default in the MISP platform. More specifically, the former modules are showing the expanded values directly on the attributes, the latter are showing and adding the expanded values via a proposal form. Moreover, the import modules add new data into MISP platform as well as export modules transmit existing data from MISP [10].

Table 11: Classification of expansion MISP modules

Module Name	Category/Type of Module	Module Operation	GitHub link
Asn history	Default expansion module (expansion and hover type)	It is expanding an AS number with the ASN description and its history.	-
Backscatter.io	Expansion module (expansion and hover type)	Expand IP address with mass-scanning observations	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/backscatter_io.py
BTC scam check	Expansion module (hover type)	Check if a BTC address has been abused	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/btc_scam_check.py
CIRCL Passive DNS	Default expansion module (expansion and hover type)	Expand hostname and IP addresses with passive DNS information.	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/circl_passivedns.py
CIRCL Passive SSL	Default expansion module (expansion and hover type)	Expand IP addresses with the X.509 certificate seen.	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/circl_passivessl.py
countrycode	Expansion module (hover type)	Gives information about the country a URL belongs to	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/countrycode.py
CVE	Default expansion module (hover type)	Gives information about a vulnerability (CVE)	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/cve.py
CVE advanced	Expansion module	An expansion module to query the CIRCL CVE search API for more information about a vulnerability (CVE).	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/cve_advanced.py
DNS	Default expansion module (simple module)	A simple module to resolve MISP attributes like hostname and domain to expand IP addresses attributes.	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/dns.py

Docx-enrich	Enrichment module	It is an enrichment module to get text out of Word document into MISP (using free-text parser).	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/docx_enrich.py
DomainTools	Default expansion module (expansion and hover type)	You can acquire information from DomainTools Whois. [12]	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/domaintools.py
EUPI	Default expansion module (expansion and hover type)	You acquire information about an URL from the Phishing Initiative project. [13]	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/eupi.py
EQL	Expansion module (expansion type)	Generate event query language (EQL) from an attribute Event Query Language. [14]	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/eql.py
Farsight DNSDB Passive DNS	Expansion module (expansion and hover type)	Expand hostname and IP addresses with passive DNS information	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/farsight_passivedns.py
Hashdd	A hover module	Check file hashes against hashdd.com [15] including NSLR dataset.	-
Hibp	A hover module	A hover module to lookup against Have I Been Pwned.	-
Intel471	Expansion module	It gets information from Intel471. [16]	-
Ipasn	Default expansion module (hover type)	Gives the capability to the system to obtain the BGP ASN of an IP address.	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/ipasn.py
Ods-enrich	Enrichment module	Get text out of OpenOffice spreadsheet document into MISP (using free-text parser).	-
PassiveTotal	Default expansion module	http://blog.passivetotal.org/misp-sharing	-
pdf- enrich	Enrichment module	Extract text from pdf into MISP (using free-text parser).	-

pptx- enrich	Enrichment module	Get text out of PowerPoint document into MISP (using free text parser).	-
sourcecache	Default expansion module	a module to cache a specific link from a MISP instance.	-
STIX2 pattern syntax validator	Expansion module	a module to check a STIX2 pattern syntax.	-
Virustotal	Default expansion module	an expansion module to query the VirusTotal API with a public key and a low request rate limit [17],[18].	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/expansion/virustotal.py
Whois	Default expansion module	a module to query a local instance of uwhois [19], for the time being the whois protocol has been replaced by this: lookup.icann[21]which subsequently affects all the other sites operating on the basis of it	-

Table 12: Export MISP modules

Module Name	Category of Module	Module Operation	GitHub link
CEF	Export module	Export information in Common Event Format (CEF)	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/export_mod/cef_export.py
GoAML export	Export module	Export information in GoAML format	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/export_mod/goamlexport.py
Lite Export	Export module	Export information a lite event	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/export_mod/liteexport.py
Simple pdf export	Export module	Export information in pdf	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/export_mod/pdfexport.py

ThreatConnect	Export module	Export information in ThreatConnect CSV format	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/export_mod/threat_connect_export.py
ThreatStream	Export module	Export information in ThreatStream format	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/export_mod/threatStream_misp_export.py

Table 13: Import MISP modules

Module Name	Category of Module	Module Operation	GitHub link
CSV import	Import module	Customizable CSV import module	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/import_mod/csvimport.py
Cuckoo JSON	Import module	Cuckoo JSON import	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/import_mod/cuckooimport.py
Email import	Import module	Email import module for MISP to import basic metadata	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/import_mod/email_import.py
GoAML import	Import module	GoAML format import	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/import_mod/goamlimport.py
OCR	Import module	Optical Character Recognition (OCR) module for MISP to import attributes from images, scan or faxes.	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/import_mod/ocr.py
OpenIOC	Import module	OpenIOC import based on PyMISP library	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/import_mod/openiocimport.py

stiximport	Import module	It gives the capability to process STIX xml/json	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/import_mod/stiximport.py
ThreatAnalyzer	Import module	It gives the capability to process ThreatAnalyzer	https://github.com/MISP/misp-modules/blob/master/misp_modules/modules/import_mod/threatanalyzer_import.py
Misp-workbench	Import module	It gives tools which transmit data out of the MISP MySQL database and utilise and modifiy them outside of MISP platform as well.	https://github.com/MISP/misp-workbench
MISpego	Import module	Maltego Transform to put entities into MISP events	https://github.com/MISP/MISPego
Misp-maltego	Import module	Set of Maltego transforms to interface with a MISP instance.	https://github.com/MISP/MISP-maltego
PyMISP	Import module	Python library using the MISP Rest API. This is the official library for MISP and can also generate offline MISP events.	https://github.com/MISP/PyMISP
MISP-STIX-Converter	Import module	A utility repo to assist with converting between MISP and STIX formats.	https://github.com/MISP/MISP-STIX-Converter
MISP-Taxii-Server	Import module	An OpenTAXII Configuration for MISP with automatic TAXII to MISP sync.	https://github.com/MISP/MISP-Taxii-Server
mail_to_misp	Import module	Connect user/infrastructure email to MISP in order to create events based on the information contained with mails.	https://github.com/MISP/mail_to_misp

Finally, MISP modules can run on the same system or on a remote server. Python 3 is a requirement for the installation and execution of MISP modules. These modules extend MISP capabilities through python scripts. The MISP modules could extend it, without any customisation and have also the capability of auto-discovery of new modules with their features.

4. CTI information flow

Cyber-threat intelligence tends to be synonym to the improvement of cyber posture of an organisation. It is not of minor importance, that NIST [20] encourages greater sharing of cyber threat information among organisations, both in acquiring threat information from other organisations and in providing internally generated threat information to other organisations. Implementing the following recommendations enables organisations to make more efficient and effective use of information sharing capabilities. Information flow schema in Cyber-Trust.

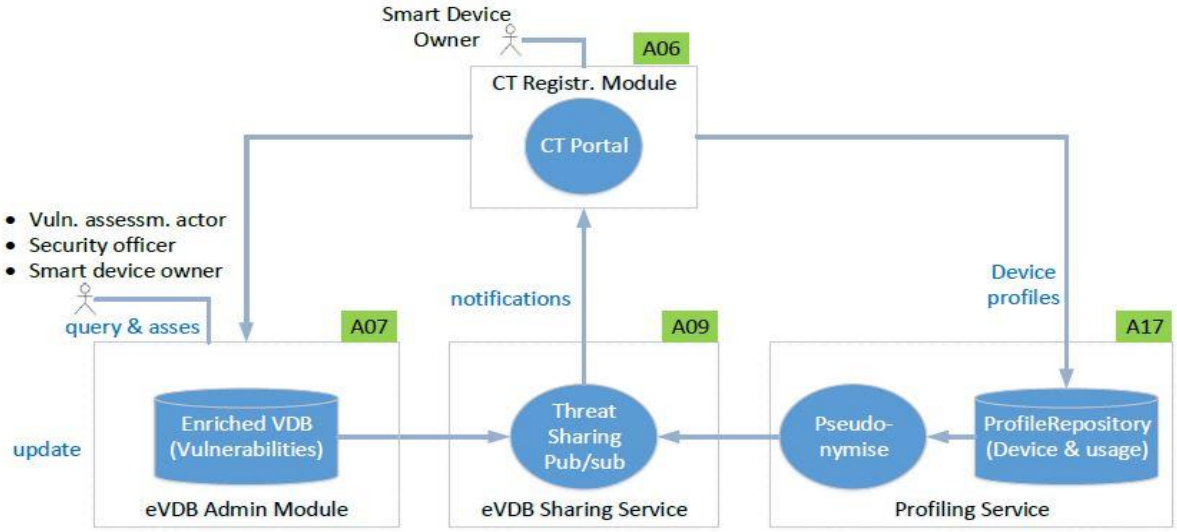

Figure 9: Information flow within Cyber-Trust

Figure 9 presents the information flow within the Cyber-Trust project. The whole cyber-threat intelligence information already exists in eVDB as well as the renewed information that comes from open sources databases are available to the users through users' subscription, Cyber-Trust registration [A06] component. MISP was adopted as the appropriate platform for storing and sharing all CTI. Finally, users (Smart-home and organisations) and subsequently their devices will have different access to information regarding their access rights to the Cyber-Trust platform.

4.1 Information sharing - eVDB Database

The eVDB Admin Module (A07) is responsible for the usage and the maintenance of the database storing enriched data which are collected through CTI techniques. The enriched vulnerability database (eVDB) admin includes information that is disseminating through the sharing service (A09) and encompasses information regarding:

- Vulnerabilities
    - CPE (Common Platform Enumeration)
    - CWE (Common Weakness Enumeration)
    - Exploits
        - Mitigation strategies for each exploitable threat
            - Type of mitigation strategy

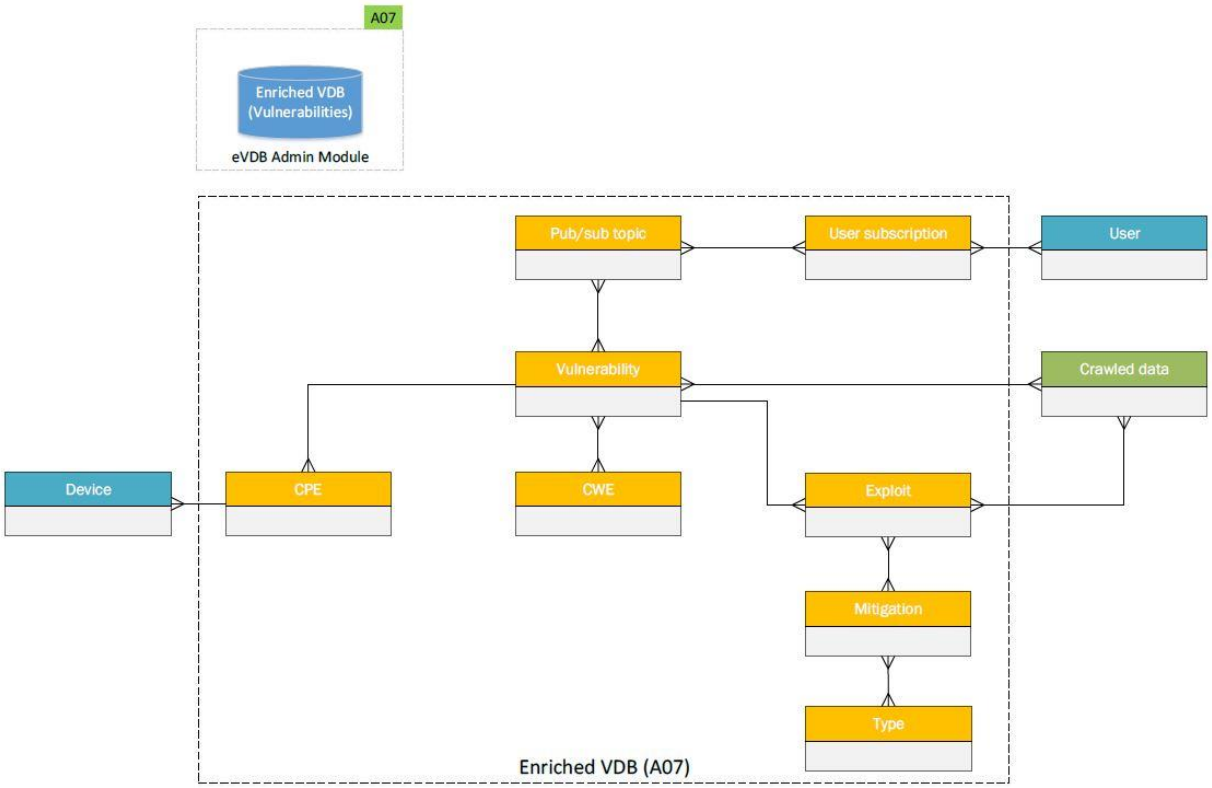

Figure 10: Data graph of Enrich Vulnerability Database [A07]

Figure 10 describes the data flow of the eVDB Admin Module. In other words, the way that the CTI is shared to the end-users and devices.

4.1.1 Flow of information to users

Regarding the Flow of information, eVDB (A07), as a primary source of data provides information regarding all the above information and capabilities that MISP provides (Section3.53) for instance the existence of vulnerabilities, decision alerting etc. and is sharing the knowledge to end-users. Cyber-Trust includes three kind of end-users, i.e.,

- Law Enforcement Agencies (LEAs),
- Internet Service Providers (ISPs),
- Smart-Home Owners (SHOs).

The disseminating knowledge is visualized in the Visualization Portal which is consisted by four (4) User Interfaces (Law Enforcement Agencies (LEAs) Interface, Internet Service Providers (ISPs) Interface, Smart Home Owners (SHOs) Interface and Administrators Interface) (see Section 0). Each user subscribes to the platform and through the pub/sub mechanism is transferred to the specific information that is available to have access.

4.1.2 Flow of information to devices

The flow of information to devices follows the architectural specification defined in D4.4.

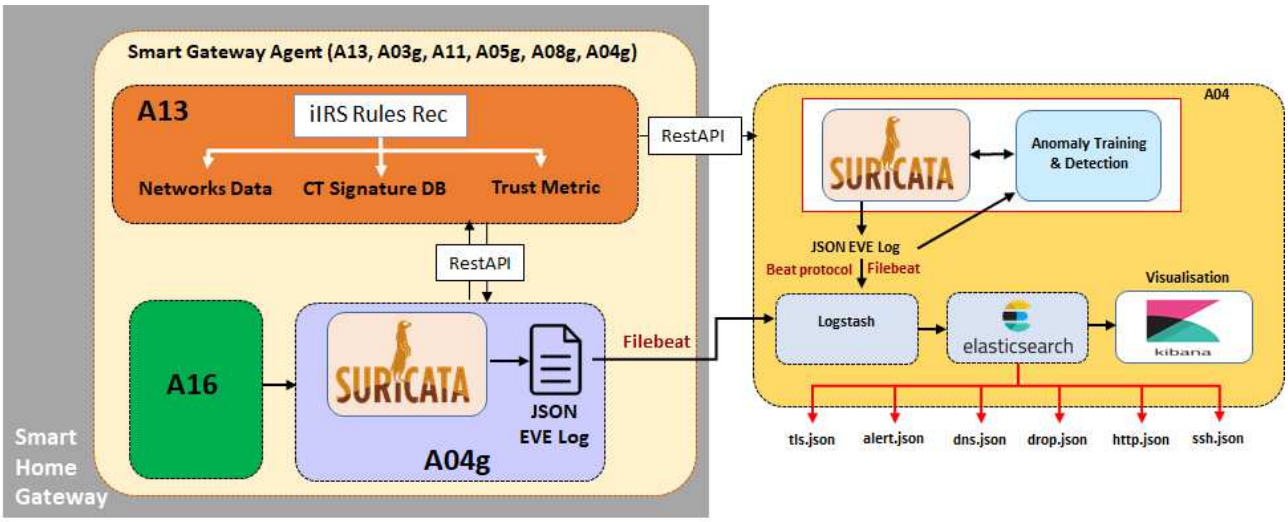

Figure 11: Components (A16, A04G, A04) responsible for flow of information in devices

There are two main components within the home environment that require interacting with the network infrastructure or/and collect traffic information, namely A16 (the network architecture and assets repository) and A04g (the intrusion detection system). Conceptually, both components may reside on the smart gateway for data collection and communication or, given the additional computational requirements, they may be relocated on a separate hardware device but closely connected to the smart gateway.
The main function of A16 is to collect information about the devices connected to the smart home network, the network infrastructure/connectivity, and the traffic exchanged between the network and the Internet. The network traffic can indeed be collected from the LAN and WAN interfaces of the smart gateway and subsequently processed for storage using NetFlow. The network infrastructure is inferred using a combination of discovery mechanisms (Nmap specifically) and querying the services on the smart gateway (from ARP and DHCP leases to VLAN and routing information). The raw traffic is also passed to the anomaly detection module for examination and identification of potential attacks within A04G.

4.1.3 Crawler

The information gathered by the Cyber-Trust Crawler (A10) is also stored in the enriched vulnerability database (eVDB) (Figure 10). Thus, eVDB is enriched with CTI discovered in social, clear, deep and dark web, including related forums, marketplaces and security-related websites. To do so it utilizes an ensemble of state-of-the-art data processing and machine learning techniques to identify the web pages that have cyber-threat intelligent information and should be crawled and to extract/contextualize all relevant threat information. This kind of information is leveraging the collected information to identify emerging threats, zero-day vulnerabilities and new exploits to IoT devices.

5. User interface

The purpose of the User interfaces within the CTI sharing module is to present the data to the end-user in a clear and precise displaying, avoiding users' confusion due to little explanatory data and, where possible, finding graphics solutions (e.g. graphs or graphic representations) to allow more immediate understanding of information.

5.1 Objectives of user interfaces

In the design and implementation of the interfaces, the main objectives are to keep the presentation to the user as clean and as clear as possible. The data collected by the CTI are intended for an audience of users

who certainly have specific knowledge, but despite this, the interfaces should maintain a clear aspect and explain the information effectively, even to a user with very high skills.

At the same time, it is important to maintain rapid access to information to ensure the users to move between the various topics without spending too much time.

The main objectives followed in the planning and implementation phase of the UI were:

- Limited number of 'clicks' to reach the various information (3 clicks, maximum 5 for particular cases).
- Intuitive interfaces even for non-specialized users.
- As compatible as possible with the used platforms, both software (browser) and hardware (desktop, mobile).

5.2 Technical specifications

The entire UI component was developed using the NodeJS platform for scaffolding and for managing the connections to the various modules, while the individual interfaces were created using the HTML and JQuery languages.

Specifically, the connections between the interfaces and the Crawler component was made using RESTful calls, while the connection to the MISP component was implemented using a Python script based on PyMisp.

In both cases, after obtaining the input data, these are managed directly on the client, which also takes care of their graphic representation.

5.3 Visualization interfaces

The data are presented in the UI trying to maximize their readability and to minimize their reading time. In case of data that are obtained in the form of a list or a list, it was decided to arrange them in a tabular form, taking advantage of the possibility of ordering and filtering the results dynamically. In the example shown in Figure 11 it is possible to see the representation in table form of the results coming from the MISP component: the selection of information was made to allow the user to quickly identify which are the records of interest, with no need to examine every single piece of information.

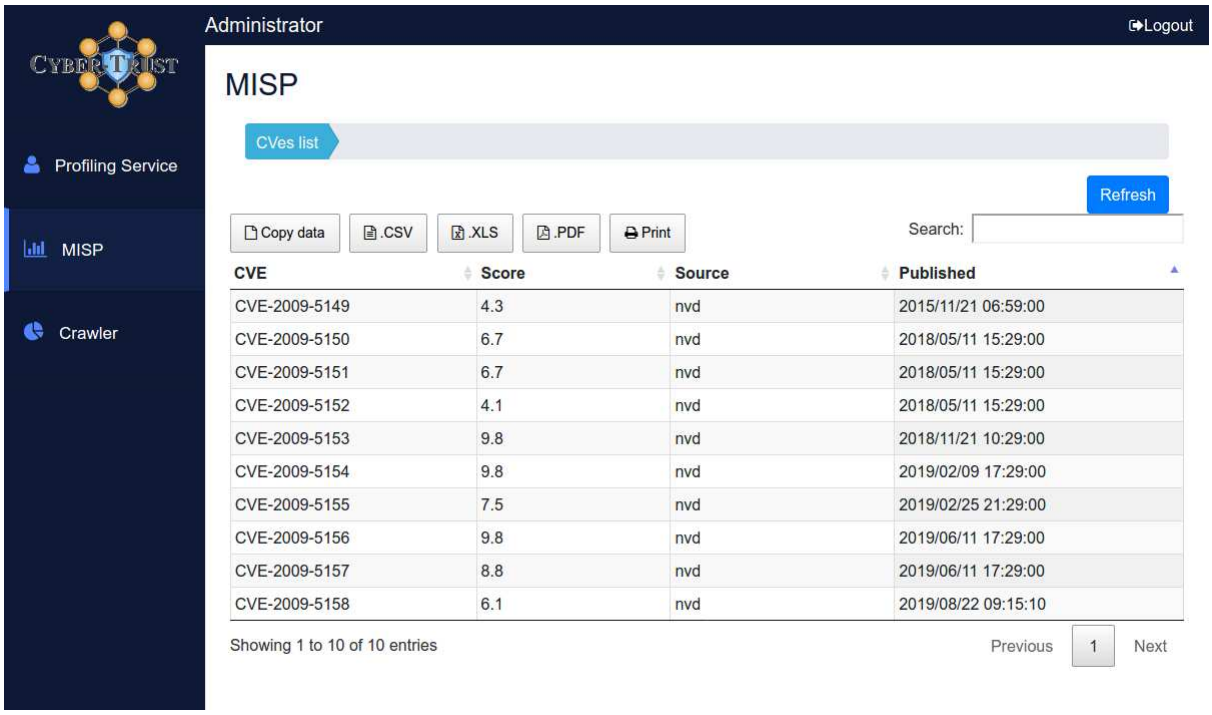

Figure 12: List of Results coming from MISP

Once a record has been selected, it is transported to the single record tab, where all the information is more fully exposed Figure 13.

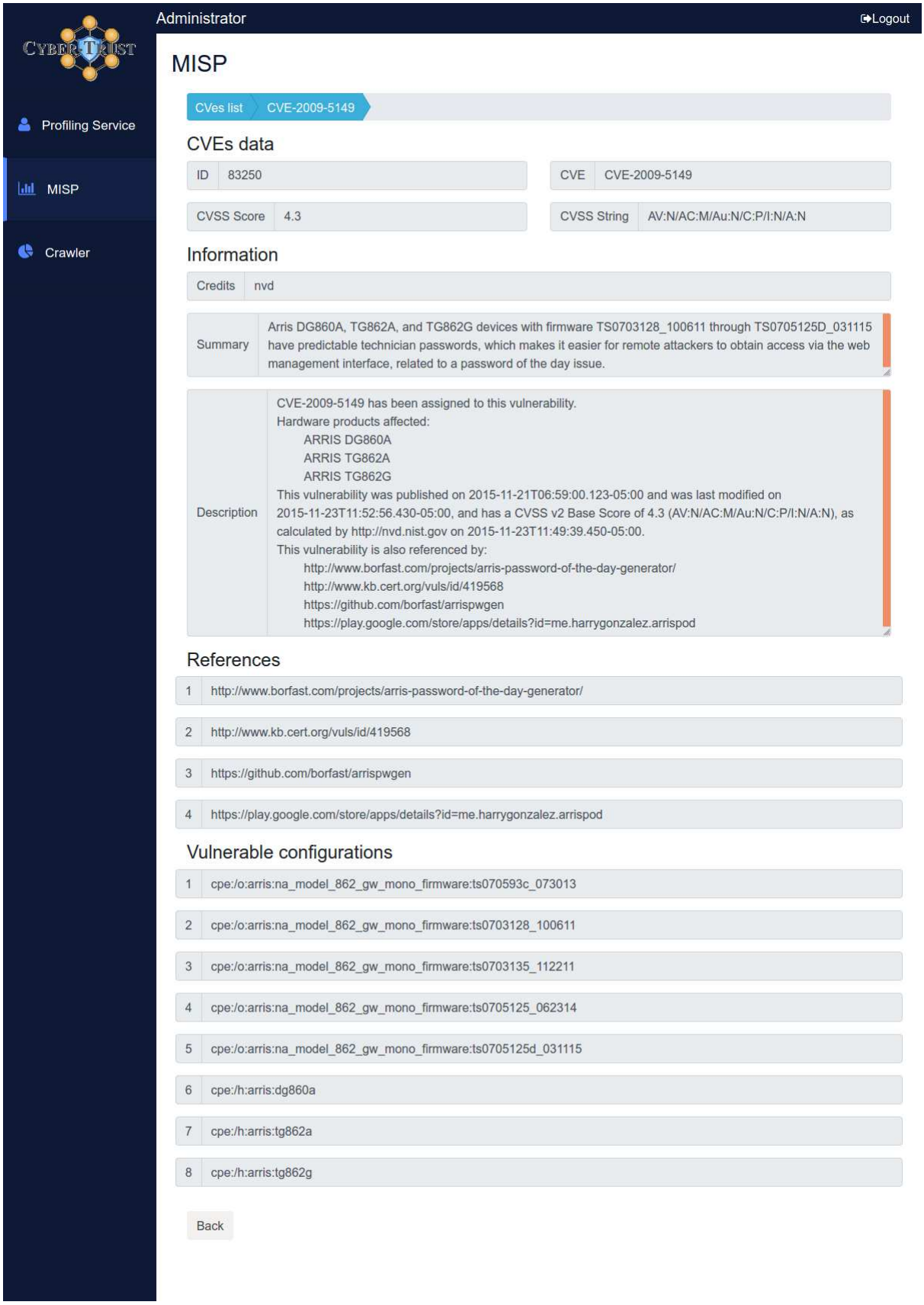

Figure 13: Single Record Tab of a result coming from MISP

If the data are lent in graphic representations (e.g. quantitative/qualitative data, time series), these are presented to the user in the graphic form.

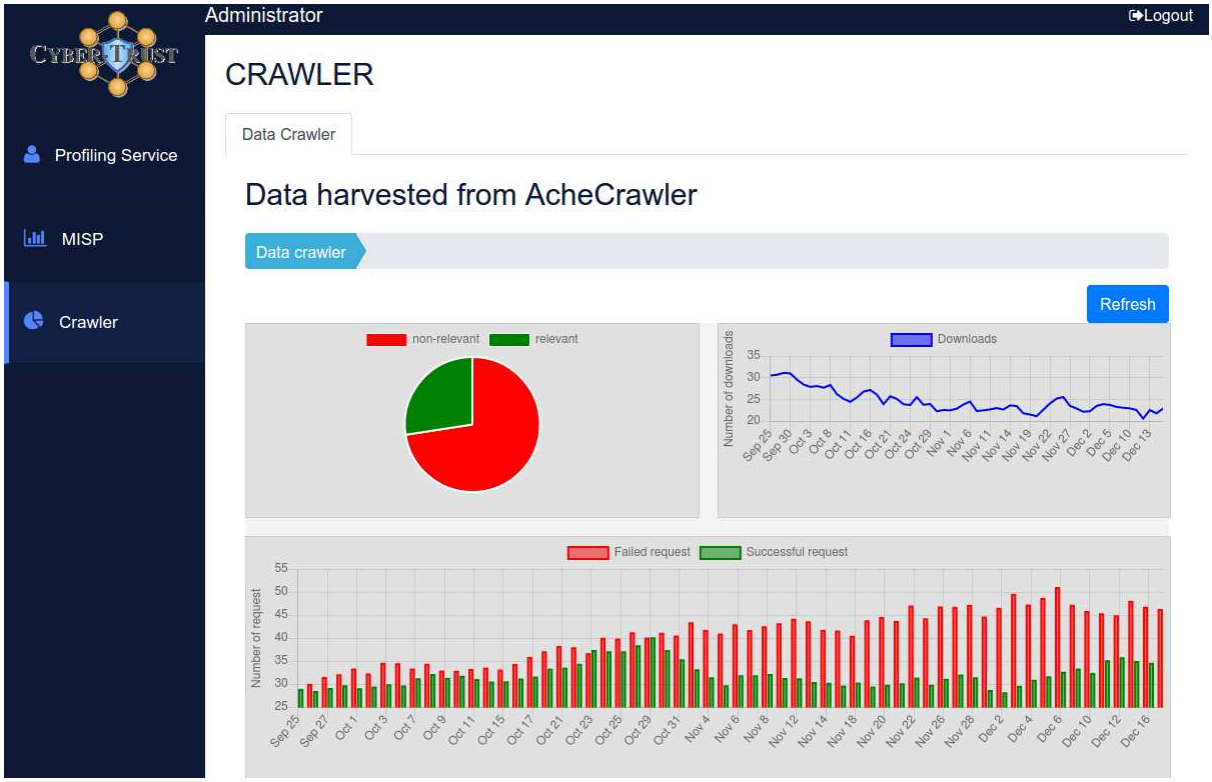

Figure 14: Graphical Representation of crawler's data

6. Legal aspects

For the creation of the Cyber-Trust information collection and storage systems, the technical partners take into account the general framework described in T3.1 (D3.1). The latter describes the challenges in the context of innovation and cybersecurity, since the project uses a number of new technologies in its various components. The produced report sheds light onto the respective regulatory framework and the legal and ethical requirements. Those requirements have been further specified, during the course of time, in the consequent T3.3 and the respective Deliverables (D3.3 and D3.4).

In particular, based on the outcomes of the first Data Protection Impact Assessment (DPIA) (D3.4), the individual assessment of the eVDB by the technical partners (independently from other components), showed that no personal data are targeted as such during the research phase and no such intention is expressed in case of a potential commercial use of the platform. However, when assessed in a holistic manner with the rest of the platform elements, as seen earlier, it becomes evident that the eVDB correlates to more components and constitutes part of a number of distinct information flows. Thus, whether personal data will be processed or not in relation to the processing operations referring to the eVDB, depends on the kind of data that will be collected through the cyber-threat intelligence sources and will be fed into it, including the Crawling Service (A10). In other words, it is important to distinguish into two data processing operations: the 'input', which constitutes the gathering of the material that flows into the eVDB and the 'output', which corresponds to the material visible to and shareable with, the members of the Cyber-Trust platform.

Concerning the input, in the first DPIA, the eVDB was assessed, exclusively with reference to the Crawling Service. The latter is another component developed for the Cyber-Trust platform, whereas the other cyber-threat intelligence sources used to feed the eVDB are external to it. Thus, with respect to the Crawling Service, the technical partners identified risks and corresponding mitigating measures. Furthermore,

regarding the possibility of incidental processing of personal data during the research phase, the opinion and guidance of the data protection officer of the lead technical partner engaged in the particular processing was sought. As for the output, this is what will be accessible via the User Interface and thus is assessed as part of the User Interface, as seen below.

With regards to the information flows to devices, given the large-scale character of the data collection, the technical partners will have to take into consideration all the legal and ethical requirements with respect to privacy and data protection, in order to avoid indiscriminate collection and retention of personal data and other information that may provide insights into an individual's personal life, and ensure security on the basis of data minimisation and data protection by design and by default. Strict criteria for the monitoring of a device should be established relating to the likelihood and severity of an attack.

As for the information flows to the end-users and the respective User Interface, the technical partners have considered various options, based on each end-user's access rights. The information accessed via the User Interface in the MISP context should avoid including personal data.

All in all, the components and the corresponding data processing operations will be re-assessed in the second DPIA (D3.5) in Month 35. In the first DPIA, the data processing operations were assessed per component. In the second, since the platform will have reached a level of maturity, the data processing operations will be assessed based on the identified information flows, including the intelligence sharing flows presented in this report.All the components will be re-assessed in the second DPIA (D3.5) in Month 35.

7. Conclusion

Undoubtedly, organisations and software applications invest great amount of resources towards to cyber-threat intelligence every year. CTI Sharing techniques are used in order to boost the security mechanisms of organisations and applications as well to enhance the knowledge in cyber security research field as a whole. Cyber-Trust has developed a cyber- threat intelligence sharing tool, in order to gather the cyber-threat information and transfer the malicious knowledge internally, to the Cyber-Trust components. A part of information is transferred also to other affiliated users and platforms. For Cyber-Trust sharing capabilities we use MISP as the sharing platform and STIX as the sharing mechanism. In a nutshell, we could say without any hesitation that cyber-threat intelligence sharing provides lots of positive impact to technological solutions. Nevertheless, it is a necessity to handle technical and legal challenges.

8. References

[1] Cyber–Trust D2.2, "Threat Sharing methods: comparative analysis," 2018.

[2] The quest for the appropriate cyber-threat intelligence sharing platform, Thanasis Chantzios, Paris Koloveas, Spiros Skiadopoulos, Nikos Kolokotronis Christos Tryfonopoulos, Vasiliki-Georgia Bilali, Dimitris Kavallieros

[3] InfoSecurity Group Magazine, [Online] available: http://infosecurity-magazine.com/opinions/cyber-intelligence-sharing/, [Accessed 03 01 2020]

[4] Sixth Annual Data Breach Preparedness Study, [Online] available: https://www.experian.com/data-breach/2019-data-breach-preparedness, [Accessed 03 01 2020]

[5] MISP, "An Introduction to Cybersecurity Information Sharing", [Online] available: https://www.misp-project.org/misp-training/0-misp-introduction-to-information-sharing.pdf, [Accessed 03 01 2020]

[6] MISP, "MISP -User Guide A Threat Sharing Platform", [Online] available: https://www.circl.lu/doc/misp/book.pdf , [Accessed 03 01 2020]

[7] MISP, "MISP Galaxy Clusters", [Online] available: https://www.misp.software/galaxy.pdf , [Accessed 03 01 2020]

[8] MISP, "MISP Objects", [Online] available: https://www.misp-project.org/objects.html, [Accessed 03 01 2020]

[9] MISP, "General Layout", [Online] available: https://www.circl.lu/doc/misp/general-layout/, [Accessed 03 01 2020]

[10] MISP, "Extending MISP with Python modules", [Online] available: https://www.misp-project.org/misp-training/3.1-misp-modules.pdf, [Accessed 03 01 2020]

[11] MITRE ATT&CK, "ATT&CK for Industrial Control Systems", [Online] available: https://attack.mitre.org/, [Accessed 10 01 2020]

[12] DOMAINTOOLS, "Turn Threat Data into Intelligence", C : http://whois.domaintools.com/, [Accessed 13 01 2020]

[13] Phishing Initiative, "EU-PI project website, European union anti-phishing initiative", [Online] available: https://phishing-initiative.eu/?lang=en, [Accessed 13 01 2020]

[14] Event Query Language (EQL), EQL, [Online] available: https://eql.readthedocs.io/en/latest/, [Accessed 13 01 2020]

[15] HASHDD, "Build and Search Treat Feeds", [Online] available: https://www.hashdd.com/, [Accessed 13 01 2020]

[16] INTEL 41, "CYBERCRIME INTELLIGENCE" [Online] available: https://intel471.com/, [Accessed 13 01 2020]

[17] Virus Total, "Basics", [Online] available: https://developers.virustotal.com/reference, [Accessed 13 01 2020]

[18] VIRUSTOTAL, "VIRUSTOTAL GUI", [Online] available: https://www.virustotal.com/gui/home/upload, [Accessed 13 01 2020]

[19] Uwhois, [Online] available: http://www.uwhois.net/, [Accessed 13 01 2020]

[20] NIST Special Publication 800-150, "Guide to Cyber-Threat Information Sharing", Computer Security, Chris Johnson, Lee Badger, David Waltermire, Julie Snyder, Clem Skorupka

[21] ICANN LOOKUP, [Online] available: https://lookup.icann.org/, [Accessed 20 01 2020]