D3.2

CERT Pilot Architecture

WP3 – CERT Pilot

C3ISP

Collaborative and Confidential Information Sharing and Analysis for Cyber Protection

Due date of deliverable: 30/09/2017
Actual submission date: 30/09/2017

30/09/2017
Version 1.0

Responsible partner: ISCOM-MISE
Editor: Sandro Mari
E-mail address: sandro.mari@mise.gov.it

Project co-funded by the European Commission within the Horizon 2020 Framework Programme		
Dissemination Level		
PU	Public	**X**
PP	Restricted to other programme participants (including the Commission Services)	
RE	Restricted to a group specified by the consortium (including the Commission Services)	
CO	Confidential, only for members of the consortium (including the Commission Services)	

The C3ISP Project is supported by funding under the Horizon 2020 Framework Program of the European Commission DS 2015-1, GA #700294

Authors: Sandro Mari (ISCOM-MISE), Andrea Saracino (CNR)

Approved by: Than Hai Nguyen (CEA), Vincent Herbert (CEA), Lukasz Sobkowiak (GridPocket)

Revision History

Version	Date	Name	Partner	Sections Affected / Comments
1.0	02/08/2017	A. Saracino, S. Mari	CNR, ISCOM MISE	ToC
2.0	05/09/2017	A. Saracino, S. Mari	CNR, ISCOM-MISE	Preliminary Architecture Description
3.0	18/09/2017	A. Saracino, S. Mari	CNR, ISCOM-MISE	Data model presentation
4.0	24/09/2017	A. Saracino, S. Mari	CNR, ISCOM-MISE	First review ready version
5.0	28/09/2017	A. Saracino, S. Mari	CNR, ISCOM-MISE	Final Version

Executive Summary

This deliverable presents the architecture of the CERT pilot. Building on the results of deliverable D3.1, in this deliverable it will be shown how the C3ISP architecture will match the pilot requirements, presenting both a high level and detailed view of the architectural component. The CERT pilot is general and imposes noticeable challenges, since, differently from other pilots, it has to be ready to receive and handle any possible type of CTI information, managing data with different format and semantic. Moreover, this pilot envisions a plurality of possible prosumers, hence the interface must be general enough to match the requests of both private users, public and private organizations of different size.

After presenting the architectural model and its relation with the ones defined in D7.2, the single components will be detailed, presenting the operations which are specific for this pilot, needed to integrate C3ISP with the specific functionalities, in particular those for data collection and dispatching. Afterward, the deliverable will delve in the data which will be used for analysis, their format, privacy requirements and desired analysis. The deliverable will close with considerations on security, requirements matching and plan of future work.

Table of contents

1. Introduction

Due to continuous raising of cyber-threats, cyber-security information sharing is a helpful practice for raising awareness, and early detection/prevention of recent and new attacks. The effectiveness of such a practice is mainly related to two factors: the timeliness of information sharing, and their utility.

Information collected by CERT might in fact be not up-to-date and generally raw and unfiltered, bringing thus large shares of data, which might be useless. After collection and filtering, the CERT should share the extracted information with the right stakeholder, being sure not to bother uninterested parties with information which are not useful for it.

Thus, while the automation is the key for the achievement of timely collection, the correct classification and its mapping to the correct stakeholders is the key to ensure utility of the shared data. An information to be shared should contain some basic issues. First of all if it is necessary to differentiate information related to an **incident,** i.e. related to a particular event recorded in the network, like the presence of malware, malicious traffic, particular checked compromising etc., or if it is related to a **threat**, i.e. a vulnerability, a notice, an high level information.

For *incidents*, it should be reported every related (Indicator of Compromise) IoC: IP affected, timestamp(s), URL(s) involved, samples etc.

For *vulnerabilities,* every useful detail for analysis should be reported: detailed description, referring CVE if exists, affected systems info etc.

For what concerns communication, in case of incident, the affected IP block is the primary key for finding the interested stakeholders. After analysis, the extracted information related to the incident should be forwarded to the victim, which is found according to a tree-like mechanism, exploiting the affected IP block itself. As first instance the Autonomous System owner, generally the abuse contact or, better, a direct contact inside the organization, should be informed. However, in particular case, the owner of the IP block should be informed directly, in order to reduce delay time for the incident solution.

For vulnerabilities, the key factor to determine the interested stakeholder could be the economic sector. For example, a SCADA vulnerability could be of interest for the energetic sector. It must be considered that, sometimes, also an incident could be seen as a threat for not involved subjects, so, it should be sent to potential affected third parties, outside of the economic domain of the vulnerability provider. It is worth noting that in such a case, data anonymization or other privacy requirements might be mandatory.

1.1. Purpose

The purpose of this deliverable is the description of the software and hardware architecture for the instantiation of the ISCOM-MISE pilot.

1.2. Scope

This deliverable will describe the inclusion of the C3ISP framework in the ISCOM-MISE native components for CTI sharing and management. We will present the specific instantiation of the C3ISP framework, which matches the requirements presented in deliverable D7.1 and its integration with the functionalities envisioned. Furthermore, we will present the type of data that will be analysed, their privacy requirements and the desirable analytics. Finally, some security requirements will be provided, together with the strategy to address them within the C3ISP framework.

1.3. Definitions and Abbreviations

Term	Meaning
API	Application Program Interface
C3ISP	Collaborative and Confidential Information Sharing and Analysis for Cyber Protection
CERT	Computer Emergency Response Team
CVE	Common Vulnerabilities and Exposures
DMO	Data Manipulation Operations
DSA	Data Sharing Agreement
HE	Homomorphic Encryption
IAI	Information Analytics Infrastructure
ISI	Information Sharing Infrastructure
TLS	Transport Security Layer

2. System Overview

An overview of the pilot architecture is depicted in Figure 1, reporting the main actors, component and expected interactions.

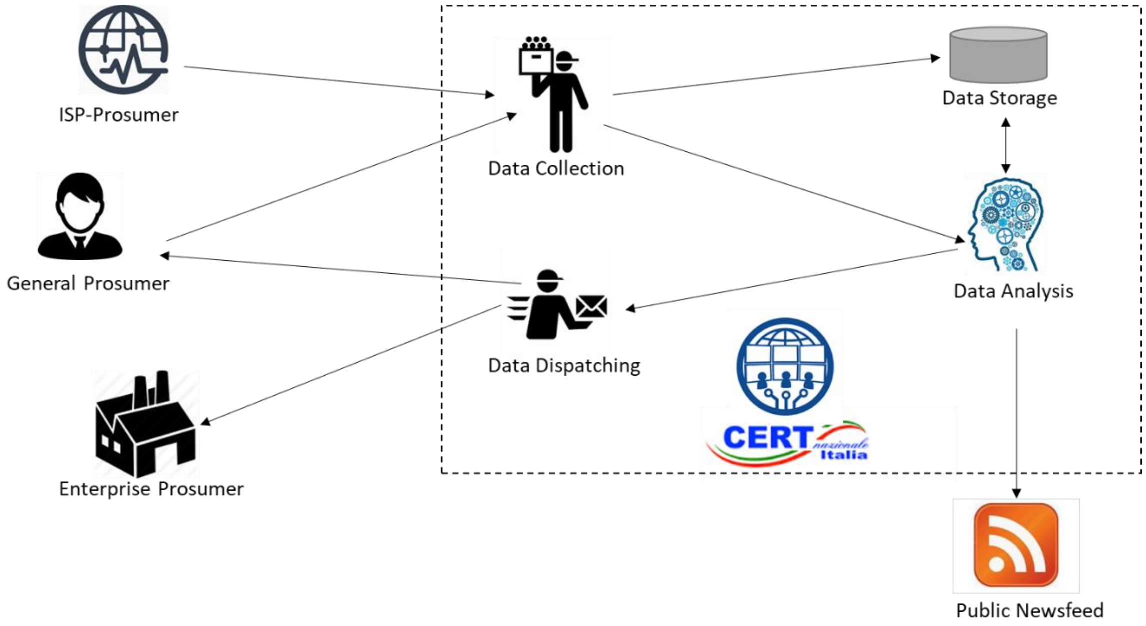

Figure 1: CERT Pilot Overview: main actors and functionalities

The CERT pilot is the one which better represents the C3ISP philosophy, including all the C3ISP functionalities in a single entity and having as prosumers a large set of stakeholders, which also includes the other pilots (i.e. ISP and Enterprise/SME). The CERT is a public entity which collects data related to cyber threats (CTI) from several prosumers (or providers), stores and categorize the collected information and exploits them to run analysis. In particular, the analysis can be requested from a specific prosumer, or issued by the CERT itself. Analysis results will be stored in the DPO as well, given that the data policy allows it, and/or are dispatched to interested prosumers. Furthermore, a set of information, collected or inferred will be made publicly available through the CERT website as newsfeed related to cyber threats of public interest.

The CERT has to consider the privacy requirements expressed by the data providers, which might apply to the data content itself or to computed results. The policies will specify which attributes can be made available in public access, a list of specific prosumers which can read data and derived analysis results and legal requirements on methodologies for data storage and processing. Moreover, this pilot requires that prosumers might be able to enforce some data policies on their premises, sanitizing data before sharing them with the CERT.

3. System Architecture

The CERT pilot architecture follows the hybrid model with On-Premises ISI with Centralised ISI and IAI, exactly as described in deliverable D7.2, section 3.2.

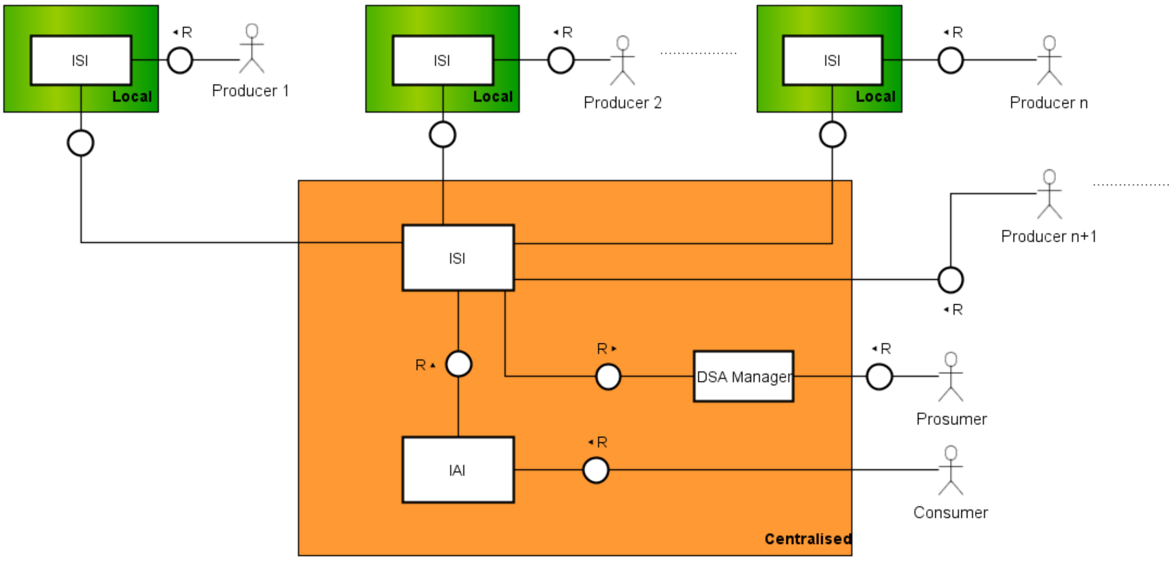

Figure 2: C3ISP Hybrid Architecture

Hence, the envisioned architecture envisions the presence of a Local ISI on prosumer side, whilst the "centralized" architectural part entirely resides in CERT premises. The presence of the local ISI also allows to accommodate the requirement, also introduced in the former section, related to the sanitization of data on prosumers premises, before they are shared with the CERT.

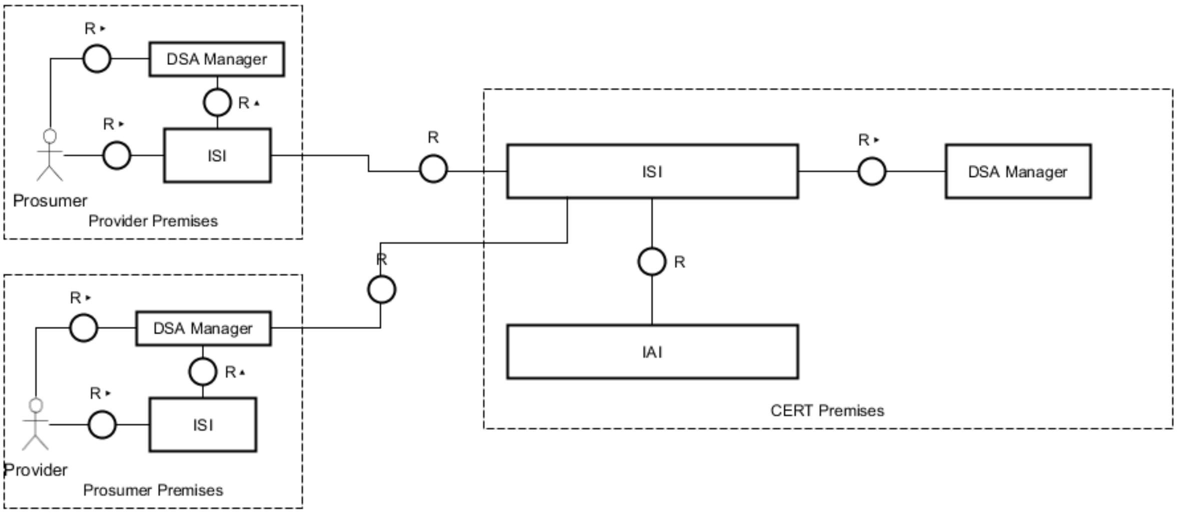

Figure 3: CERT Pilot Architecture

A high-level representation of the architecture is depicted in Figure 3, showing the components and their interconnections. As discussed, the ISI is present on both the Prosumer and CERT premises. In the following they will be addressed respectively as *Local ISI* and *Remote ISI.* An instance of the DSA manager is present both on provider/prosumer premises and in the CERT. The DSA manager local to prosumers is used, as in the other pilots, to define policies for the shared data. Part of these policies will be enforced by the local ISI, additional policies will be enforced instead by the remote ISI, in particular the one related to data analysis and to result redistribution. The remote DSA manager is instead used to define additional constraints which are internal to the CERT organization, which, being a public organization has to implement standards related to data storage and maintenance. These policies are enforced by the remote ISI. The IAI is only present in the CERT premises, hence the prosumers are considered not able to run in house the analytics on their data and will demand the analysis directly to the CERT. Hence, the C3ISP analytics functionalities are all provided by the CERT, on prosumer request, or invoked directly by the CERT itself.

4. Component Architecture

This section will detail the components of the CERT pilot, describing their functionalities, the performed operations and their interactions.

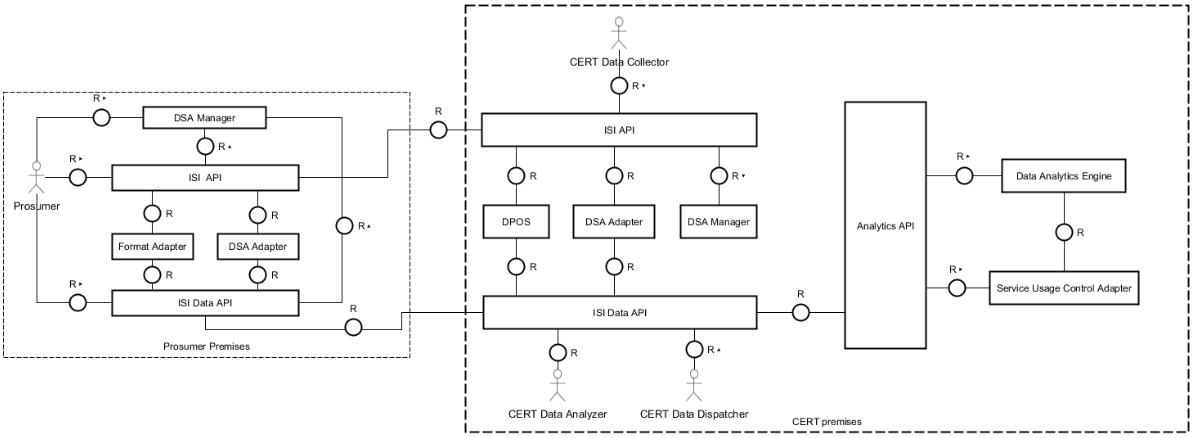

Figure 4: Detailed pilot architecture

The detailed pilot architecture, also showing the interaction with the main actors internal to the CERT, i.e. the data collector, data analyser and data dispatcher, as shown in deliverable D7.1 is presented in Figure 4. Also, the pictures show more in details the fact that the Analytic components and the DPOS only reside on the CERT side. We will now discuss the single components in details.

4.1. *Block Design*

In the following we will report the block representation and the description of the main blocks of the CERT pilot architecture. In particular, we will describe the local architecture Prosumer/Provider side of the C3ISP infrastructure, and the C3ISP architecture for the CERT side.

4.1.1. The Prosumer local side

The operations and components related to the prosumer are detailed in Figure 5.

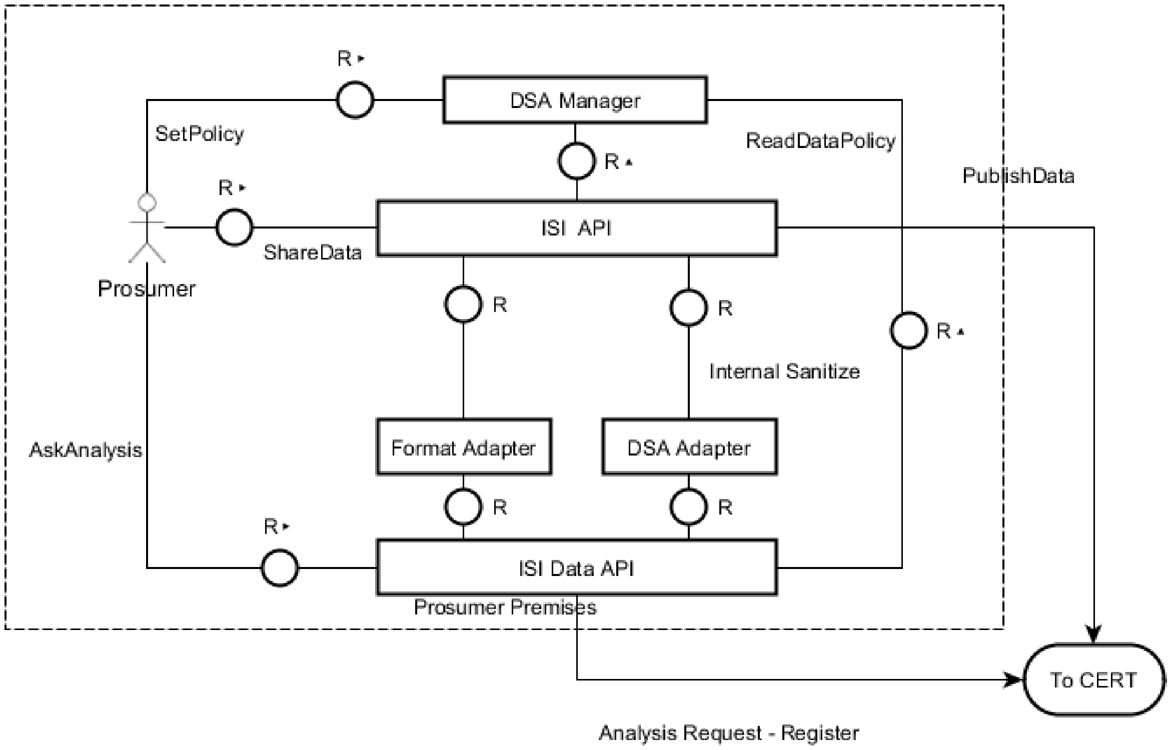

Figure 5: Prosumer side detail

In the CERT pilots the Prosumer will mainly do either:

i. asking the CERT for specific analytics
ii. or sharing CTIs with the CERT to let it compute additional knowledge about specific threats.

As shown, the prosumer interacts directly with the DSA manager to define policies for data to be shared, which might be partially enforced on the prosumer side by the DSA adapter, which will provide to sanitize the data before they are sent to the CERT.

Hence, the prosumer operations are the following:

- Ask Analysis - Register: The prosumer queries the CERT for specific data and/or for an analysis (classification, clustering etc.) on dataset already stored in the CERT DPOS. The AskAnalysis might also result in a Registration to a specific topic or set of information.
- PublishData: The prosumer introduces a piece of data, already structured according to the CTI format standard, protected through a data bundle with the attached DSA. According to the DSA, data can be sanitized before they are sent to the CERT, which will store the bundle in the DPOS.
- InternalSanitize: Is the operation of sanitization performed by the DSA adapter local to the prosumer.

We assume thus that in the CERT pilot the prosumer is not able/willing to run in house analytics and will demand this task to the CERT itself.

4.1.2. CERT Remote Side

The CERT side includes the majority of the components of the pilots. In Figure 6 is reported a more detailed view of the CERT infrastructure in the C3ISP pilot,

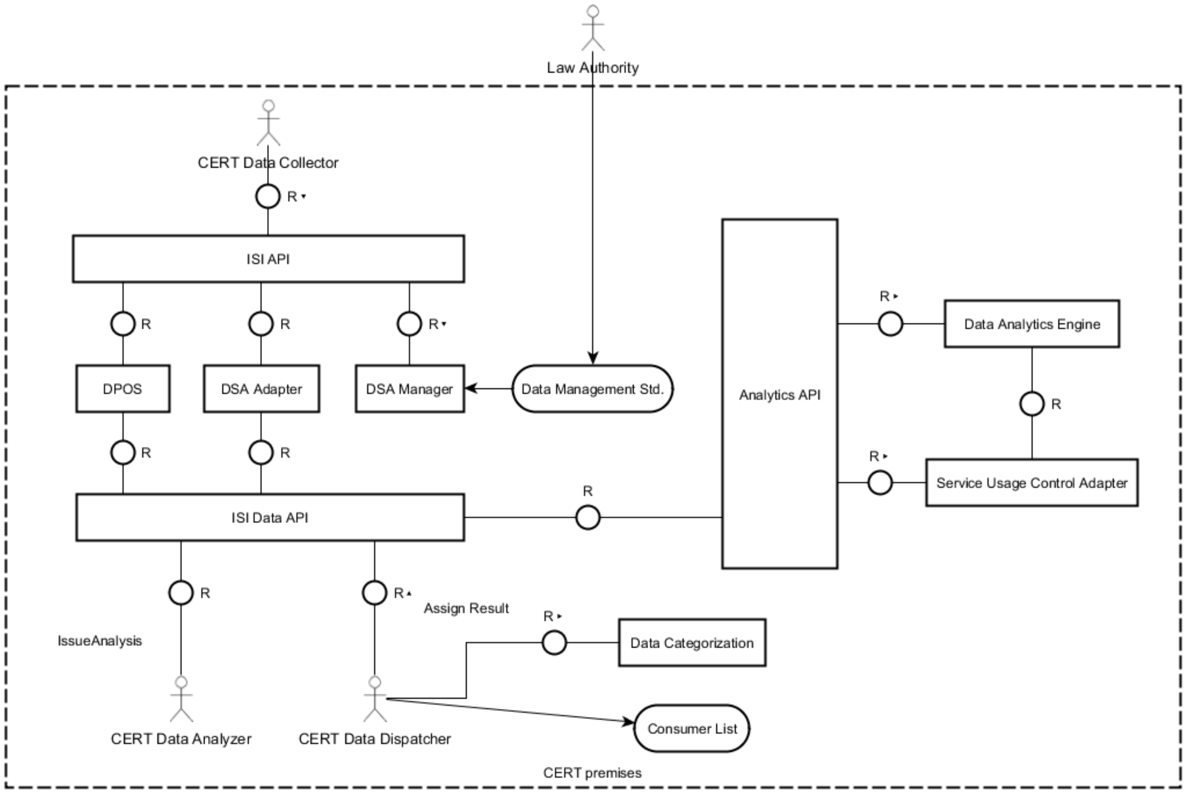

Figure 6: Detailed architecture of the CERT side

The three main actors on the CERT side are the Data Collector, the Data Analyzer and the Data Dispatcher. The Data Collector directly interacts with the ISI API, managing thus the data storage operations acting as interconnection between the Local and Remote ISI. The Data Collector might either passively act by receiving and storing data from prosumers in the DPOS, or actively act by requesting specific information or data streams from prosumers. The Data Analyser issues the analysis operations, either when receiving requests from prosumers, or acting as a consumer itself, generally to infer information of public interest. The Data Dispatcher interacts with the ISI via API for receiving the analysis results which have been extracted by the IAI. It also receives the registration requests issued by prosumers, storing them in a Consumer List. Hence, through the CERT internal Data Categorization component, the dispatcher matches the analysis results with the consumer registrations, sending automatically interesting results to them, without the necessity for them to issue an analysis. It is worth noting that such a process is done automatically, providing results coming from collaborative analysis without explicit user analysis request. Finally, it is worth noting the presence of a set of Data Management Standards, which are used to define additional policies for data, enforced on the CERT side. These standards are defined by law authorities and might regard national or international regulations for data storage, management of classified information, etc.

The functions to be considered on the CERT side are:

- IssueAnalysis: Invoked by the Data Analyzer by itself, or as a response to a prosumer request for a specific analytic.
- AssignResult: This function is invoked by the data dispatcher to match a new information, received by a prosumer or computed through the IAI, with interested

consumers. The information dispatching phase is however handled by the ISI, to ensure that result is delivered in accordance with DSA policies.

4.2. Performed Analytics

The CERT is intended to perform data analysis on a large set of possible data types, with different syntax and semantic. The introduction of C3ISP already improves the current workflow by the introduction of the format adapter, which handles the different formats, building thus standard CTI records. For the CERT pilot, will be considered three data types on which analysis will be performed.

1. Spam emails: undesired emails collected through honeypots, which might be vector for malicious code, phishing campaigns and other security attacks.
2. Malware Samples: Unclassified malware samples collected from honeypots as well. Detecting the kind of malware and the performed behaviour might be useful for early identification of new threats.
3. Botnet Connection Logs: a set of logs extracted from different providers which can be used to classify IPs in order to differentiate between static and dynamic IPs, allowing a better categorization.

In the following will be discussed the data format, the intended analysis and the privacy requirements, with a preliminary identification of the necessary DMOs.

4.2.1. Input Data

We will report here details on the kind and format of data, as they are given by prosumers, i.e. before they are processed from the local ISI. This description will provide the basics for the definition of heuristics for the format adapter presented in D7.2.

4.2.1.1. Spam Emails

Spam emails are one of the best known and most annoying issue on the internet. Spam emails cause several problems, spanning from direct financial losses, to misuses of Internet traffic, storage space and computational power. At the same time, spam emails are an effective tool to perpetrate different cybercrimes, such as phishing, malware distribution, or social engineering-based frauds.

The spam emails come as *eml* files, which is a structured document intended to be read and interpreted by email clients, still it can be read with any standard text editor. An example of eml is reported in the following.

```
Return-Path: <calendario22@seminarios-empresariales.com>
Received: from smtp.iit.cnr.it (mx4-local [192.168.1.151])
        by mx6.iit.cnr.it (Cyrus v2.3.7-Invoca-RPM-2.3.7-16.el5_11) with LMTPA;
        Fri, 22 Jul 2016 18:49:02 +0200
X-Sieve: CMU Sieve 2.3
Received: from localhost (localhost [127.0.0.1])
       by smtp.iit.cnr.it (Postfix) with ESMTP id E6DDDB81270
       for <andrea.saracino@iit.cnr.it>; Fri, 22 Jul 2016 18:49:02 +0200 (CEST)
X-Virus-Scanned: Debian amavisd-new at mx4.iit.cnr.it
Received: from smtp.iit.cnr.it ([127.0.0.1])
```

```
        by localhost (mx4.iit.cnr.it [127.0.0.1]) (amavisd-new, port 10024)
        with ESMTP id bwZaDJ07-Pht for <andrea.saracino@iit.cnr.it>;
        Fri, 22 Jul 2016 18:49:00 +0200 (CEST)
X-SMTP-Auth: no
Received: from mail3.seminarios-empresariales.com (mail3.actualizacionesenlinea.com [204.12.217.117])
        by smtp.iit.cnr.it (Postfix) with ESMTP id 8D5E8B8144A
        for <andrea.saracino@iit.cnr.it>; Fri, 22 Jul 2016 18:48:59 +0200 (CEST)
Received: from WIN-9RVGNEAE8GB (204.12.217.115) by mail3.seminarios-empresariales.com id
hi979g0our09 for <andrea.saracino@iit.cnr.it>; Fri, 22 Jul 2016 11:49:25 -0500 (envelope-from
<calendario22@seminarios-empresariales.com>)
X-NEB: Gen-000973
Message-ID: <15435fda0a2a3bbac2086e5000122bbb@seminarios-empresariales.com>
From: "=?utf-8?Q?Capacitaci=C3=B3n_Online_en_Vivo?=" <calendario22@seminarios-empresariales.com>
To: <andrea.saracino@iit.cnr.it>
Subject: =?utf-8?Q?Calendario_del_mes_de_Agosto_-_En_L=C3=ADnea?=
Date: Fri, 22 Jul 2016 11:49:22 -0500
MIME-Version: 1.0
Content-Type: multipart/alternative;
        boundary="----=SPLITOR00A_001_50362114D"

This is a multi-part message in MIME format.

------=SPLITOR00A_001_50362114D
Content-Type: text/plain;
        charset="utf-8"
Content-Transfer-Encoding: quoted-printable

Buenos d=C3=ADas

Le hacemos llegar la programaci=C3=B3n de Agosto para que decida los temas =
que necesita usted y su equipo de trabajo=2E
```

As shown, the eml reports a set of typical information of emails in a well-structured format which simplifies the definition of a CTI structure and its semantics. The data reported are summarized in the next table.

Table 1: E-Mail data format and description

Element Name	Element Type	Required	Description
Sender Address	Structured String	Yes	The email address of the sender.
Recipient Address	Structured String	Yes	The email address of the recipient.

Date Received	Timestamp	Yes	Time when the email has been received.
Subject	String	No	Email subject
Body	String	No	Email body with, text, links and images.
Attachment	Binary	No	Attached files to the email.

4.2.1.2. Malware Samples

Malware are the most common attack vectors for threats on any kind of system. The daily increase in samples and type is exponential [2]. The malware samples considered in the CERT pilot are extracted from different honeypots, but they are not assigned to any specific signature. Hence, an analysis is desired in order to automatically identify the type of threat. An example of Hexdump and of malware log metadata are reported in the following box.

```
{"meta_data":{"id":382431926,"report_id":"59be265b77656223cc008d5a","ip":null,"domain":null,
"asn":null,"country_code":null,"tld":null,"api_key_id":518,"reported_at":"2017-09-
17T07:38:03.541Z","status":"NEW"},"report":{"confidence_level":0.5,"report_category":"eu.acd
c.malware","report_type":"[WEBSITES][HORGA][GARR] AMUN hexdump capture (in partnership with
ISCTI)","reported_at":"2017-09-
17T07:38:03Z","sample_b64":"FgMBpAGgAwP1A2W2w2sL6Cp4eOh+wSeOct/uiwKnpVkWRY41UJ4PJBITFRYyMzg5
QGZnamuen6KjzKoBUwUFAQoIBhcYGQsCAQ0mJAYBBgMGAgUBBQMFAgQBBAMEAgMBAwMDAgIBAgMCAgEBAQMBAv8BAQ8B
ARIVAwECAg==","source_key":"malware","source_value":"e903ad6307e42a8c38174cc2c1b89573bb69928
ef52087eddc41df306483619c","timestamp":"2017-09-
17T07:37:11Z","version":1,"report_id":"59be265b77656223cc008d5a"}}

timestamp: 2017-09-17T07:37:11Z
source_ip: 141.212.122.48
sha2_checksum: e903ad6307e42a8c38174cc2c1b89573bb69928ef52087eddc41df306483619c
cc: US
```

The list of attributes which can be extracted from a malware are reported in the following table.

Table 2: Malware data format and description

Element Name	Element Type	Required	Description
Hexdump/Binary	Text	Yes	Base64 encoding of hexdump or binary downloaded by the honeypot.
IP Source	String	No	The source IP from which the sample has been received.
Timestamp	Timestamp	Yes	Time on which the sample has been received.

4.2.1.3. Botnet Samples

Botnets are networks of devices under the control of an attacker, generally used to perform DDoS attacks, or to send spam emails in hideous way. The identification of the botnet structure and of the botmaster is a challenging task, which would benefit by the collaborative analysis on shared botnet connection logs. The logs used in the CERT pilot are extracted from different providers, such as Spamhaus [1], and are structured according to the format shown in the next box.

```
*************************** 1. row ***************************
id: 7718
ip: 109.113.82.30
asn: AS30722
cc: IT
lastseen: 1446677830
timestamp: 2015-11-04 22:57:10
botname: p2pzeus
domain: n/a
remote_ip: 54.83.43.69
remote_port: 80
local_port: 57386
protocol: tcp
firstseen: 1446677830
```

Table 3: Botnet data format and description

Element Name	Element Type	Required	Description
Id	Integer	Yes	The ID of the record
IP	String	Yes	The source IP of the connection.
ASN	String	Yes	Identifier of the autonomous system from which the connection has been originated.
CC	String	Yes	The country code of the ASN.
Lastseen	Unixtime	Yes	Timestamp of the last connection attempt (unixtime)
Timestamp	Timestamp	Yes	Timestamp of the last connection attempt
Firstseen	Unixtime	Yes	Timestamp of the first connection attempt (unixtime)
Botname	String	No	Name of the identified bot.
Domain	String	No	The domain related to the specific botnet sinkhole
Remote IP	String	No	IP related to the specific botnet sinkhole.
Local Port	Integer	Yes	Port on which the connection has been started.
Remote Port	Integer	Yes	Port on which the connection has been directed.
Protocol	String	Yes	Transport level protocol

4.2.2. Analytics operation

It is possible to envision a large set of operations to be performed on these three data types, which are representative of a large share of security threats in the wild. In the following we will report a set of selected analysis operations, divided by the specific data type.

4.2.2.1. Analytics for Spam emails

[AFS01] Campaign Clustering: emails are analysed by observation of structural and semantic features and clustered in groups by similarity. Entropy based divisive clustering are the kind of algorithms that prove to be effective for this specific analysis, in particular the CCTree algorithm presented in [3]. The rationale is that, spam emails with similar structure are generally part of the same spam campaign, hence generated by the same spammer. Such an analysis might be relevant to identify the spammer and/or botmaster.

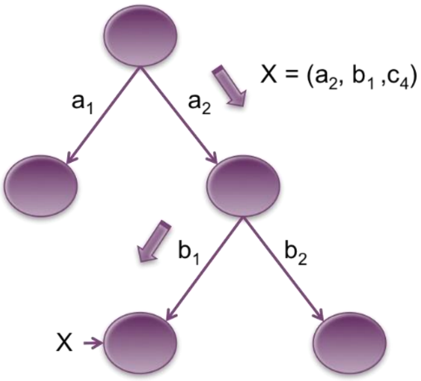

Figure 7: A CCTree

[AFS02] Campaign Classification: This analysis generally takes place after the AFS01, to assign a label, representing the kind of threat, to a specific campaign. As shown in [4] spam emails can be grouped in five classes: *advertisement, phishing, confidential trick, malware, portal.* This analysis exploits tree based classifiers to assign to each email its specific class. It is worth noting that this methodology requires a large and representative dataset for training, and the classifier needs to be continuously updated with new knowledge.

[AFS03] Link Extraction and Analysis: Links and active contents in an email are extracted and matched against databases of known dangerous domains. This technique allows to find the relation among a single email or a campaign of similar emails (eventually extracted through AFS[AF01) and a domain which could be under the control of an attacker.

[AFS04] Keyword Analysis: The email body is matched against a set of keywords which might be typical of a certain threat, for early detection of malicious activities. Since the full text of the email will be analysed, this operation might pose serious privacy issues.

4.2.2.2. Malware Analytics

[AFM01] Signature Base Analysis: The malware sample is tested against one or more antivirus to verify if a database entry for such a specific signature is already known. Effective against known viruses, still quite easy to deceive and totally ineffective against zero day threats.

[AFM02] Static Code Analysis: The malware sample is dissected through reverse engineering, to extract execution flows, execution n-grams and dangerous APIs. The analysis can be done heuristically or by exploiting machine learning tools.

[AFM03] Dynamic Code Analysis: The malware is executed in a sandbox to observe and extract its execution patterns, which are then analysed by machine learning tools. This approach is generally very effective on the side of malware detection, but imposes a considerable overhead.

4.2.2.3. Botnet Analytics

[AFB01] IP extraction and matching: the source IP and ASN are matched against other records or spam related activities for early detection of an attacker. Possible match with IPs coming from logs of the ISP pilot.

[AFB02] IP filtering and translation: Detection of patterns of mutable IP address, to detect subnets, private IPs, nat and other non-canonical patterns for IP assignment.

4.2.3. DMO Operations

As the considered data carry privacy sensitive information, a set of data manipulation operations have to be taken in consideration in order to preserve privacy. As discussed, these operations might be enforced on prosumer side, by the local ISI, or on CERT side by the remote one, according to what is specified in DSAs.

[DMOS01] Anonymize email: This operation hides am email address, which generally is the recipient address of whom the privacy should be preserved, still keeping the format useful for analysis. The rationale is that spam email DB should not be used as a source of private email address. An example of anonymized email is: *foo@foo.fo*.

[DMOS02] Suppress Text: For privacy reason the text is suppressed and only the email header can be used for analysis.

[DMOS03] Encrypt Text: The email body is encrypted to be analysed only through Homomorphic Encryption (HE) techniques. Word search, word occurrence and other measures will thus still be possible without disclosing any private content of the email.

[DMOB01] Hide IP: The IP address is anonymized by still keeping the format so that in the analysis will be possible to understand that the analysed string is actually an IP address. Example of anonymized IP is: *xxx.xxx.xxx.xxx.*

[DMOB02] Delocalize IP: The IP address can be used to track the exact location of a device initiating the connection (if public). If such an operation is specified in the DSA, the source IP address is changed with the one of the ASN, which will still allow a localization, but coarse grained.

5. Security Model

As discussed, the CERT pilot follows the Hybrid Model presented in D7.2, hence a communication over the Internet is expected to happen among the prosumers and the CERT. On such a connection the basic security properties must be ensured.

5.1. Confidentiality

Data must be sent through a secure channel and attackers should not be able to eavesdrop message while in transit between prosumers and CERT. This property is ensured by the usage of TLS and additionally by the Bundle construct used to move and store data.

5.2. Integrity

Data cannot be modified while in transit. In particular, it is important to keep the validity of the DSA. To ensure this property, the bundle is digitally signed by the prosumer when attaching the DSA. Every modification will require a new signing process. Furthermore, the integrity will be also supported by the TLS.

5.3. Authentication

The authentication is handled through the Identity Manager. Each prosumer wishing to access the CERT services, has to ask for credentials, used to access all the C3ISP services. The CERT will however make public a set of computed results which can be accessed without the necessity of authentication through a public portal.

5.4. Authorization

The authorization is handled through the ISI components acting with the DSA, in particular the DSA Adapter and the Service Usage adapter. These components will ensure that information is accessed only by authorized parties, applying the policies to the CERT itself which might not be considered trusted by some prosumers. Moreover, it will be checked that the party issuing an operation is effectively authorized to perform it. Policy evaluation considers also mutable attributes, which introduces the possibility of revoking the right to execute actions even during their execution.

6. Deployment Model

6.1. Hardware Requirements

6.1.1. Hardware Requirements for the CERT

In the following, the hardware requirements to achieve a proper and efficient running of the pilot. It is worth noting that the CERT is supposed to process continuously a large amount of information.

- *Processors*: 4 Intel/AMD 64-bit (8 cores, if provided as Virtual Core)
- *Minimum RAM*: 16 GB
- *Hard Disk*: 2 TB

6.1.2. Hardware Requirements for prosumers

Requirements for prosumers will depend by the prosumer type, which might range from private users to large companies. The minimum requirements will be the following.

- *Processors*: 2 Intel/AMD 64-bit (4cores, if provided as Virtual Core)
- *Minimum RAM*: 4 GB
- *Hard Disk*: 200 GB

6.2. Software Requirements

The CERT will require Ubuntu Desktop LTS 16.04 and/or Windows 10.

The functionalities offered by the CERT will be queried through rest APIs which will be made available by the CERT itself, in particular by the Data Collector and the Data Analyzer.

7. Requirements Matrix

Use Case	Description	Component
CERT-UC-01	It refers to the possibility of the CERT to collect and query information from stakeholders without violating privacy.	CERT Data Collector
CERT-UC-02	It refers to the possibility to perform analysis on data automatically through ML and Statistic tools.	CERT Data analyzer
CERT-UC-03	It refers to the possibility of sharing new inferred information with the right consumer, matching result policies.	CERT Data dispatcher

8. Conclusion and Future Work

In this document, we have described the logical architecture of the CERT pilot, focusing on the chosen architectural model, explaining the interactions among components, defining the operations specific of this pilot, the data type, the expected analytics and needed DMOs.

This deliverable shows that all the requirements presented in D3.1 are addressed by the presented architecture, which ensures the possibility to collect data, perform analysis and dispatch results in line with the policies specified for data and operations.

In the following months the architecture will be actually implemented and maturated, following the guidelines from WP7 and WP8 and gradually integrated in the CERT workflow. This process is expected to be completed at month 26.

9. Bibliography

[1] Spamhaus website, www.spamhaus.org.

[2] Kaspersky, INTERPOL, Mobile Cyber Threats - Kaspersky Lab and INTERPOL Joint Report, Tech. rep., Kaspersky and INTERPOL. URL http://media.kaspersky.com/pdf/965 Kaspersky-Lab-KSN-Report-mobile-cyberthreats-web.pdf.

[3] M. Sheikhalishahi, A. Saracino, M. Mejri, N. Tawbi, F. Martinelli, Fast and Effective Clustering of Spam Emails based on Structural Similarity, FPS 2015.

[4] M. Sheikhalishahi, A. Saracino, M. Mejri, N. Tawbi, F. Martinelli, Digital Waste Sorting: A Goal-Based, Self-Learning Approach to Label Spam Email Campaigns, STM 2015.