R2D2

Reliability, Resilience and Defense technology for the griD

D5.1 – Design of the Prevention Systems For Energy Infrastructures Security

Date:

30/09/2023

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them. Horizon Europe Grant

Deliverable details

Title	WP	Version
Design of the Prevention Systems For Energy Infrastructures Security	5	1

Contractual delivery date	Actual delivery date	Delivery type*
30/09/2023	29.09.2023	Report

*Delivery type: R: Document, report; DEM: Demonstrator, pilot, prototype; DEC: Websites, patent fillings, videos, etc; OTHER; ETHICS: Ethics requirement; ORDP: Open Research Data Pilot.

Author(s)	Organisation
Aggeliki Zapalidi	CYBER
Argyris Makrygeorgou	CYBER
George Aslanidis	CYBER
Kostas Papadatos	CYBER
Kostas Rantos	CYBER
Tilemachos Valkaniotis	CYBER
Anja Korošec	ELEK
Jurij Curk	ELEK
Bojan Mahkovec	ELPROS
Tadeja Babnik	ELPROS
Neven Nikolić	EMSS
Srđan Subotić	EMSS
Lucas Pons	ETRA
Ugo Stecchi	ETRA
Margo Raja	GUARD
Mihkel Väljaots	GUARD
Priit Anton	GUARD
Tanel Ojalill	GUARD
Dimitrios Stratogiannis	HEDNO
Kanellos Grigorios	HEDNO
Kontopoulos Theofanis	HEDNO
Papadimas Victoras	HEDNO
Selimis Dimitrios	HEDNO
Álex Alhambra	S2
Luis Búrdalo	S2
Sergio Villanueva	S2
Dušan Prešić	SCC

Author(s)	Organisation
Ismar Sinanović	SCC

Version	Date	Person	Action	Status*	Dissemination**
V0.1	23/05/2023	Ugo Stecchi	Table of Content	Draft	CO
V0.2	16/06/2023	Mihkel Väljaots	Product description and approach	Draft	CO
V0.3	30/06/2023	Mihkel Väljaots	Overview of PRECOG	Draft	CO
V0.4	14/07/2023	Aggeliki Zapalidi, Argyris Makrygeorgou, George Aslanidis, Kostas Papadatos, Kostas Rantos, Tilemachos Valkaniotis, Anja Korošec, Jurij Curk, Bojan Mahkovec, Tadeja Babnik, Neven Nikolić, Srđan Subotić, Lucas Pons, Ugo Stecchi, Margo Raja, Mihkel Väljaots, Priit Anton, Tanel Ojalill, Dimitrios Stratogiannis, Kanellos Grigorios, Kontopoulos Theofanis, Papadimas Victoras, Selimis Dimitrios, Álex Alhambra, Luis Búrdalo, Sergio Villanueva, Dušan Prešić, Ismar Sinanović	Background and innovation	Draft	CO
V0.5	28/07/2023	Aggeliki Zapalidi, Argyris Makrygeorgou, George Aslanidis, Kostas Papadatos, Kostas Rantos, Tilemachos Valkaniotis, Anja Korošec, Jurij Curk, Bojan Mahkovec, Tadeja Babnik, Neven Nikolić, Srđan Subotić, Lucas Pons, Ugo Stecchi, Margo Raja, Mihkel Väljaots, Priit Anton, Tanel Ojalill, Dimitrios Stratogiannis, Kanellos Grigorios, Kontopoulos Theofanis, Papadimas Victoras, Selimis Dimitrios, Álex Alhambra, Luis Búrdalo, Sergio Villanueva, Dušan Prešić, Ismar Sinanović	Relevant Use Cases and Actors	Draft	CO
V0.6	11/08/2023	Aggeliki Zapalidi, Argyris Makrygeorgou, George Aslanidis, Kostas Papadatos, Kostas Rantos, Tilemachos Valkaniotis, Bojan Mahkovec, Tadeja Babnik, Lucas Pons, Ugo Stecchi, Margo Raja, Mihkel Väljaots, Priit Anton, Tanel Ojalill, Álex Alhambra, Luis Búrdalo, Sergio Villanueva	Tool descriptions	Draft	CO
V0.7	01/09/2023	Aggeliki Zapalidi, Argyris Makrygeorgou, George Aslanidis, Kostas Papadatos, Kostas Rantos, Tilemachos Valkaniotis, Bojan Mahkovec, Tadeja Babnik, Lucas Pons, Ugo Stecchi, Margo	Final tool descriptions	Draft	CO

		Raja, Mihkel Väljaots, Priit Anton, Tanel Ojalill, Álex Alhambra, Luis Búrdalo, Sergio Villanueva			
V0.8	08/09/2023	Mihkel Väljaots	Implementation and deployment plan	Draft	CO
V0.9	19/09/2023	Lucas Pons, Ugo Stecchi, Srđan Subotić, Neven Nikolić	Internal review	Final	CO
V1.0	23/09/2023	Mihkel Väljaots	Finalization	Submitted	PU

*Status: Draft, Final, Approved, Submitted (to European Commission).

Dissemination Level: **PU: Public; **CO**: Confidential, only for members of the consortium (including the Commission Services)

Executive Summary

This document is part of the R^2D^2 project, covering contributions from WP5 (Prevention Systems For Energy Infrastructures Security) M7 – M24. The document represents the first iteration of the work conducted in WP5, which brings together technology providers and pilot partners with the goal of developing and testing innovative solutions that provide cyber-security enhancements for pilots. D5.1 covers the theoretical phase of WP5, serving as the baseline for the more practical iteration delivered in D5.2 (Final Version of the Prevention Systems For Energy Infrastructures Security). The work conducted in WP5 started in parallel with WP2 (Project Foundations), which defines the methodology of use cases, requirements and the architecture of tools. While WP2 focuses on project foundations and analysis of work conducted in technical work packages, the current document is focused on tools. WP5 delivers six tools that are integrated into pilots' infra systems with the focus of automating and increasing cyber-security related functionalities of Electrical Power and Energy Systems (EPES). In addition, a sixth tool is delivered that provides self-assessment for grid participants and thereby helps to plan cyber-security related development and activities. All these tools are delivered under a product called PRECOG. PRECOG is basically a toolkit composed by loosely coupled tools developed in tasks T5.1–T5.5. The tools described in this document (and practically delivered at the end of the R^2D^2 project) are as follows:

- **KSI tool** – developed and implemented by GUARD, tested in use cases UC36 (additionally in UC27)
- **Tokenization tool** – developed and implemented by GUARD, tested in use cases UC37 and UC38
- **CARMEN tool** – developed and implemented by S2, tested in use cases UC33 and UC34 (additionally in UC10 and UC27)
- **Sandbox tool** – developed and implemented by CYBER, tested in use case UC27
- **Upgraded UniFusion platform** developed and implemented by ELPROS, tested in use cases UC7, UC10 and UC11 in Slovenian pilot
- **Self-assessment tool** – composed by CYBER, tested in use case UC28

In addition, this document describes preliminary ideas for user interfaces as input for the second iteration, where development of user interfaces starts in task 5.6 (Integration and UI). Work conducted in WP5 is also input for WP7 (Integration, Demonstration and Validation) M13 – M36. [1] This deliverable, together with the other technical deliverables D3.1, D4.1 and D6.1, contributes to achieve the Milestone #3 “Design of the four Products”, due by M12. As a matter of fact, each of these documents describes in detail the design of the product to which it refers along with the methodology and techniques used.

Keywords

KSI tool, Tokenization tool, CARMEN tool, Sandbox tool, UniFusion platform, Self-assessment tool, cyber-security, deployment plan.

Copyright statement

1 Table of Contents

1 Table of Contents 7

1.1 List of tables 8
1.2 List of FIGURES 8
1.3 Acronyms and abbreviations 9
2 Introduction 11

2.1 Purpose and scope of the Document 11
2.2 Structure of the Document 11
3 Background and approach 12

3.1 Overview of the product 12
3.2 State of the Art 13
3.2.1 Background 13
3.2.2 Innovation provided 16
3.3 Relevant Use Cases and Actors 19
3.4 Approach 23
4 Product Description 26

4.1 Task 5.1 – Identification and authentication of energy IoT and edge devices 29
4.1.1 KSI tool 29
4.2 Task 5.2 – Energy tokens and trading certificates security 31
4.2.1 Tokenization tool 32
4.3 Task 5.3 – cyber-security Events Management tools 34
4.3.1 CARMEN tool 35
4.3.2 UniFusion platform 42
4.4 Task 5.4 – Deep learning data analytics for security 45
4.4.1 CARMEN tool in UC34 46
4.5 Task 5.5 – Device origin and supply chain 51
4.5.1 Sandbox tool 52
4.5.2 Self-assessment tool 64
4.6 Implementation and deployment Plan 73
5 Conclusions and next steps 77

6 References 78

8. ANNEX I 79

1.1 LIST OF TABLES

Table 1 – Acronyms and abbreviations 9

Table 2 – Pilot's partners and assets in **WP5** 13

Table 3 – Use cases and actors 19

Table 4 – Tool-related requirements 21

Table 5 – Approach used to organise work in WP5 24

Table 6 – Supply Chain guidelines 68

Table 7 – Deployment plan of second iteration 76

1.2 LIST OF FIGURES

Figure 1 – PRECOG tools and related pilots 27

Figure 2 – UC36 component layer diagram 30

Figure 3 – UC37 component layer diagram 32

Figure 4 – UC38 component layer diagram 33

Figure 5 – UC33 component layer diagram 35

Figure 6 – CARMEN main dashboard 38

Figure 7 – UC33 component layer diagram for Serbian pilot 40

Figure 8 – UC33 component layer diagram for Greek pilot 41

Figure 9 – UC33 component layer diagram for Slovenian pilot 42

Figure 10 – UniFusion platform concept 43

Figure 11 – Communication gateway with UniFusion platform 43

Figure 12 – Example of data flow as used in UC10 44

Figure 13 – UC34 component layer diagram 46

Figure 14 – UC34 component layer diagram for Serbian pilot 49

Figure 15 – UC34 component layer diagram for Greek pilot 50

Figure 16 – UC34 component layer diagram for Slovenian pilot 51

Figure 17 – Sandbox tool process flow 53

Figure 18 – Sandbox tool architecture components 54

Figure 19 – Staging environment high level architecture 58

Figure 20 – Sandbox tool User authentication 62

Figure 21 – Sandbox tool new component registration 62

Figure 22 – Sandbox tool Component Validation & Classification 63

Figure 23 – Sandbox tool Classified Components' List 63

Figure 24 – Self-assessment tool process flow 66

Figure 25 – Self-assessment tool Architecture Components 67

Figure 26 – Self-assessment tool users' authentication 70

Figure 27 – Self-assessment tool results dashboard 71

Figure 28 – Self-assessment tool List of available assessments 71

Figure 29 – Self-Assessment Tool questionnaire 72

1.3 ACRONYMS AND ABBREVIATIONS

Table 1 – Acronyms and abbreviations

Abbreviation	Definition
APT	Advanced Persistent Threat
CARMEN	S2 tool for anomaly detection and threat hunting.
CGM	common grid model
CLAUDIA	S2 tool for anomaly detection and threat hunting on windows endpoints
CYBER	Technology provider – CYBER NOESIS IKE
DB	Database
DSO	Distribution system operator
EDPS	Demo partner – EDP España S.A.U.
ELEK	Distribution System Operator – ELEKTRO LJUBLJANA PODJETJE ZADISTRIBUCIJO ELEKTRICNE ENERGIJE D.D.
ELPROS	ELPROS ELEKTRONSKI IN PROGRAMSKI SISTEMI DOO
EMSS	Transmission system coordinator – EMS SERVICES DOO
EPES	Electrical Power and Energy System
ETRA	Technology provider – ETRA INVESTIGACIÓN Y DESARROLLO S.A.
FIREWALL	Device used to block or deny malicious communications
GUARD	Technology provider – Guardtime OÜ
GUI	Graphical user interface
HEDNO	Distribution System Operator – DIACHEIRISTIS ELLINIKOU DIKTYOU DIANOMIS ELEKTRIKIS ENERGEIAS AE
HV	High voltage
ICCS	INSTITUTE OF COMMUNICATION AND COMPUTER SYSTEMS
IDP	Identity Provider
IGM	Individual grid model

Abbreviation	Definition
INTELLIGENCE UNITS	Pseudocode created by the analyst that defines and correlates data to get alerts in the system
INTELLIGENT cyber-security MODULE	CARMEN module for threat detection based on Machine Learning
LDAPS	Lightweight Directory Access Protocol over TLS
LV	Low voltage
ML	Machine Learning
RCC	Regional coordination centre
RES	Renewable energy resources
S2	Technology provider – S2 GRUPO DE INNOVACION EN PROCESOS RGANIZATIVOS SL
SAML	Security Assertion Markup Language
SCADA	Supervisory Control and Data Acquisition
SCC	Security Coordination Centre – CENTAR ZA KOORDINACIJU SIGURNOSTI SCC DOO BEOGRAD-VOZDOVAC
SGAM	Smart Grid Architecture Model
SIEM	Security information and event management
TLS	Transport Layer Security
TSO	Transmission system operator
UI	User interface
UniFusion	ELPROS Multifunctional system for telemetric applications
VM	Virtual Machine

2 Introduction

2.1 PURPOSE AND SCOPE OF THE DOCUMENT

This document describes the tools delivered in work package WP5 (Prevention Systems For Energy Infrastructures Security) and delivers vital input for the next phase of WP5 that will be described in document D5.2 (M13 – M24). This document delivers a deployment plan (design) for tools that are components of the PRECOG product. Tools covered in this document serve two specific objectives defined in R^2D^2 project – contributing to the improvement of the overall security and resiliency in power system (SO1) and increasing the cyber-security and cyber-resilience in OT and IT of the EPES (SO3) [2]. Furthermore, this document presents the work started in WP2 (Project Foundations) that is relevant for tool development and is continued in WP5 where the definition and selection of relevant use cases with pilot partners is conducted, as well as providing the design of architecture for the tools.

2.2 STRUCTURE OF THE DOCUMENT

Section one includes a list of figures, tables, acronyms and abbreviations used in the document.

Section two introduces the purpose and structure of this document.

Section three provides general information of PRECOG product, which is a collection of cyber-security related tools developed and tested under WP5. Furthermore, section three brings together tools, technology providers, pilots and relevant actors which all are represented in 10 use cases. Also, section three describes the approach used in WP5 to organise work and deliver results.

Section four is dedicated to tasks T5.1–T5.5, which describe tools in more detail. Functionality, architecture, resources and user interface are keywords to follow. Section four ends with a deployment plan which is input for the second iteration of R^2D^2 project.

For conclusion, the substantive part of the document ends in section five, which concludes work which has been conducted under WP5 and guides the reader to relevant topics, which need attention when the next iteration of R^2D^2 begins.

Section six is dedicated to references.

3 Background and approach

This section introduces the PRECOG product and the technology providers who will develop and test cyber-security related tools in WP5. This section provides a list of hardware and software assets involved in use cases that are potential part of pilot's infrastructures. Also, the innovation of GUARD, S2, CYBER and ELPROS, is discussed here. Furthermore, the following section covers 10 use cases that help to define a path for pilots and technology providers for development planning.

While defining and reviewing use cases started in WP2 (Project Foundations), the work with use cases continued in WP5. The analysis of use cases can be read from D2.1 (Requirements and Detailed Architecture Design) [3].

Under this section, the list of tool-related requirements is delivered by tool providers and pilot partners, as the result of the work carried out in WP2 and WP5.

Lastly, this section proposes the approach that is mandatory for WP5 participants to follow during work conducted under WP5.

3.1 OVERVIEW OF THE PRODUCT

The relevance of cyber-security related tools is rising in parallel with the automation of the energy grid as legacy (analogue) assets are replaced with new digital (IoT) devices. PRECOG (Prevention Systems For Energy Infrastructures Security) is one of four products delivered in the R^2D^2 project. PRECOG focuses on describing and designing cyber-attack detection and prevention tools for EPES assets. Under the PRECOG six different tools are eventually developed and implemented on pilot sites.

The CARMEN tool (developed and implemented by S2) provides two AI-based functionalities: traffic characterization and pattern detection. The KSI tool (developed and implemented by GUARD) provides data integrity through unique signatures. The Tokenization tool (developed and implemented by GUARD) provides trust of data through data tokenization. The Sandbox tool (developed and implemented by CYBER) provides monitoring of newly developed components.

Furthermore, the Sandbox tool combines the CARMEN tool and the KSI tool as additional modules for the increase of cyber-security. In addition, a Self-assessment tool (produced by CYBER) provides analytical cyber-security assessment through the format of questionnaire. The UniFusion upgraded platform (developed and implemented by ELPROS) will provide secure communication protocols using the latest protective communication measures such as data encryption and access filtering.

These six aforementioned tools can be selected from the pool of PRECOG tools separately or in combination to serve the specific needs of the client.

3.2 STATE OF THE ART

This subsection introduces the technological background of tools with a list of assets provided by pilots for developing and testing the tools. Additionally, the technology providers GUARD, CYBER, S2 and ELPROS are also introduced, and the innovation of their tools is explained.

3.2.1 Background

Table 2 presents both the hardware and digital assets with available data from pilots that can be given to the technology providers in the R^2D^2 project. All six tools developed under WP5 provide the pilots an opportunity to test novel cyber-security related functionalities that could be used in future grid solutions.

Table 2 – Pilot's partners and assets in WP5

Pilot	Pilot's partner	HW assets	SW assets	Data
Greek	HEDNO	• AMI system and smart meters on LV, MV customers and RES • SLAM meters • Staging environment in collaboration with ICCS (see attached file for description)	• SCADA/DMS • Outages Monitoring and Reporting System • Duplicate of real systems for testing (server infrastructure)	• Real-time consumption and production data from telemetered LV, MV customers and RES • Load curves for non-telemetered customers based on historical data
Serbia	SCC, EMSS	• SCC SCADA system (UC33) • Server available to install CARMEN (UC33, UC34)	• SCADA	• Real-time information provided from SCADA • IGMs and CGMs • Maintenance / outage plans collected from different stakeholders (TSO, DSOs, and PPs). • Access to real time data from IT network (UC33, UC34) • In a long term, access to SCADA network when

				available (UC33, UC34) • 1 day of PCAP traffic from IT network and/or SCADA network to pre-analyse data.
Slovenia	ELEK	• Smart meters • RTUs • DER • Servers with virtual machines	• ADMS/SCADA • Metering/billing system • Flexibility server	• Measurements from metering system • Measurements from SCADA • Network topology (CIM) • Data from DER
Spain	EDPS		• SCADA	• Location and layout (GIS) • Images • Voltage and current

GUARD

Energy grid operation relies on data that is being continuously measured by various devices and frequently collected to analyse current grid state and forecast energy demand for the near future (e.g. 15 minutes). This data is critical to enable safe and efficient grid operation (energy production and transfer from producers to consumers) and even more so for energy transfers across borders capitalising on maximum possible transfers to generate revenue for both the energy producers and for the transferrers. Although such data collection and processed data sharing is limited to private networks due to regulations. On the other hand measures to secure data before transfers from IoT to collection and processing or to cross-border is rather limited where transfer protocol is using secure http or tcp connection utilising TLS. This leaves data susceptible for modifications wherever it is stored temporarily or for long periods of time without having any mechanisms to validate it.

GUARD participates in R^2D^2 with the confidence that the KSI tool and the Tokenization tool can be implemented to improve electricity grid data cyber-security that is being collected, processed, measured and stored within and between DSOs, TSOs and RCCs. Core technology of these tools, blockchain technology, has been used to register data on blockchain for later data validations by relying on widely distributed and immutable proofs that are not susceptible for modifications nor require secrecy or management of private keys. It enables data validation starting from its creation up to its usage by third parties to whom it has been shared to whilst having full confidence on data integrity, authenticity and registration time throughout the data lifecycle. By applying this technique on EPES data would help to enhance data security within organisations and in cross-border scenarios.

CYBER

An important aspect in the energy grids cyber-security is the supply chain cyber-security and its associated risks that are addressed by multiple standards and best practices. Guidelines like ISO/IEC 27036, ISO 20243:2018, Federal Energy Regulatory Commission (FERC) and

others. [4] [5] [6] Converged IT/OT environments, however, pose major challenges when it comes to applying conventional cyber-security risk management, due to a significantly modified threat landscape and the vulnerabilities found in, traditionally, isolated legacy systems. Moreover, such methodologies are typically standalone solutions that do not interact with other security solutions, and therefore, they do not utilise any feedback that would be valuable to the risk assessment process. As such, they do not provide an environment to assess dynamically newly identified threats and vulnerabilities to provide an advanced situational awareness system that keeps monitoring the overall security posture.

Having identified the lack of integration of the existing solutions, CYBER went one step further to introduce a new tool – called The Sandbox tool – capable of assessing and classifying the operation of a newly acquired software or a software update in an isolated environment, before deploying it to the production environment. Although such a process – to test and monitor the operation of a new software in a non-production environment – is not a not known procedure, CYBER will integrate all the distinct components in a unified tool capable to assist the entire process and additionally share the results to the EPES community.

As for the Self-assessment tool, currently CYBER uses the available documents (standards and guidelines) from security institutes and organizations as references and assesses the cyber-security risk of the Supply Chain management using scorecards implemented in MS Excel files. For the PRECOG's needs, CYBER will develop the Self-Assessment Tool - to strengthen EPES and their vendors supply chain capabilities providing a central point of reference with the available guidelines and standards and the required questionnaires to assess their current status.

S2

Threat Hunting analysis, anomaly detection and machine learning (ML) anomaly detection related experience are represented under T5.3 and T5.4, covered by S2 Grupo. S2 Grupo is a European cyber-security and critical information infrastructure protection (CIIP) company driven by an engineering and expert team whose mission is to help society and its stakeholders to mitigate the new technological and cyber risks. Since S2 establishment, in 1999, S2 Grupo has become a leading company in the security sector at the national level. Currently, the company has offices in Valencia, Madrid, Barcelona, Brussels, Lisbon, Bogota and Mexico City. Since 2007, S2 has its Security Operation Centre, S2 Grupo CERT FIRST, providing managed security services globally to private and public organizations, including several governmental CERT. Among S2 Grupo's customers are leading domestic and international companies of the banking and insurance sector, Government, energy, industry and distribution sectors.

S2 Grupo's developments within the scope of R^2D^2 will be related to CARMEN [7], the tool developed by S2 Grupo together with Spain's National Cryptologic Centre, to identify compromises by Advanced Persistent Threats (APTs). CARMEN is a SIEM (Security Information and Event Management) tool with specific capabilities aimed at threat hunting. CARMEN, like other SIEMs, is a tool that collects, processes, and analyses information to generate intelligence mainly from real-time analysis of security alerts generated by applications, network traffic, etc. It is made up of agents that compile traffic flows (collection elements), a database engine where information is inserted and a web application that allows representing and checking the collected information so that analysts can work on it and make decisions based on the results provided by the tool.

ELPROS

UniFusion platform is used for large scale telemetric systems as WAMS/WAMPAC and universal telemetric systems in power systems. UniFusion platform main functionalities are:

- Complex communication gateway which supports all standard protocols for power system automation. Some of them are: IEC 61850, IEC 60870-5-101/104, DNP 3, ICCP/TASE.2, OPC-UA, IEEE C37.118, etc. Another set are protocols for messaging data exchange as MQTT, AMQP, etc.
- CIM functionality
- Complex fast database
- System for real-time processing with wide set of complex functions for real-time processing.
- System for data visualisation in windows forms format and html formats for Internet browsers.
- Systems for high reliability implemented as automatic redundant systems.
- Platform enables system extending with additional toolboxes for specialised applications.

UniFusion system is used as telemetric system in ELEK as a part of Flexibility system. UniFusion tool can improve electricity grid cyber-security for DSOs and TSOs.

3.2.2 Innovation provided

Data integrity protection has not been used so far in the process of individual grid model (IGM)/common grid model (CGM) exchange among TSOs and RCCs who are using these to characterise and adjust grid settings to optimise power delivery across different regions and within specific regions. In the R^2D^2 project T5.1 (UC36), two project partners (SCC and EMSS from Serbia) in collaboration with technology provider (GUARD) are testing KSI tool which provides data integrity validation for IGM/CGM. IGMs are created by the transmission system operator (TSO) to characterise the current grid state, it is then forwarded to and consumed by the regional coordination centre (RCC) who would then create CGM based on collected IGMs to forecast next period and distribute the models to TSOs and other RCCs for them to adjust the grid for optimum power delivery and balancing. In every step (data creation/storage and transfer) it is important to be absolutely sure that the data is in its original state and has not been changed since its creation. Subchapter 4.1.1 describes the functionality and architecture of the KSI tool in more detail.

Tokenization of energy assets and related data both on a local and global scale is still in its experimental phase, as blockchain technologies are relatively new and there is still lots of legacy hardware in use (TSOs and DSOs are actively digitalizing energy grids to increase automation), which is a limiting factor. With the automation of the energy grid, tokenization of grid-related data becomes possible. In the R^2D^2 project (T5.2), the novel approach of tokenization is tested with two pilots (HEDNO from Greece and ELEK from Slovenia) in collaboration with a technology provider (GUARD). With HEDNO work is conducted based on UC37 (Energy data tokenization), where the tokenization tool provides data immutability verification for energy consumption data which is generated by smart meters. Tokenization

tool is also tested with ELEK to ensure that grid balancing data can be trusted on a granular level before usage. The collaboration with ELEK is based on UC38 (DSO grid balancing data tokenization). Tokenization tool is explained in more detail in subchapters 4.2.1 (Tokenization tool – Energy data tokenization) and 4.2.2 (Tokenization tool – DSO grid balancing data tokenization).

ELPROS UniFusion platform will provide a complex communication system that collects data from various data sources. Communication resources can be exposed to the threat of a cybersecurity attack, which can endanger the supply chain. An innovative approach with real-time threat detection can prevent major security consequences with rapid detection.

Developments in T5.3 of the R^2D^2 project will aim at developing new capabilities for data ingestion and threat detection for CARMEN, as well as at improving the existing ones of the tool. These developments will include the development of new specific protocol dissectors for CARMEN, such as MQTT, ICCP 60870-6/TASE.2, IEC 60870-5-104 or Modbus, as well as new pre-processing and aggregation capabilities that help to reduce the amount of information to be processed or its inner variability. As a result of these developments, it will be possible for CARMEN to carry out a more in-depth analysis of network traffic at different levels and to improve its detection capabilities, and to develop new ones, both signature-based and anomaly-detection-based.

Developments in T5.4 of the project aim at developing new capabilities for the detection of new (zero-day) threats and APT threats. APTs present a significant security challenge, even for organisations with robust security measures and up-to-date practices. These threats, often orchestrated by experienced and highly skilled attackers, have the potential to inflict severe damage on numerous systems, including the sensitive IT/OT environment within which EPES operate. The inner nature of APTs makes them difficult to detect and, since they are usually tailored to a specific target, it is not possible to build a straight detection rule which raises an alarm each time they appear.

When dealing with APTs, it is common to rely on anomaly detection methods and techniques which, attending to different features, learn or model the behaviour of a system in normal circumstances, so that an alarm is raised when the observed behaviour of the system is too different from the expected one, previously learned or modelled. Anomaly thresholds, which determine when an observed behaviour is considered normal or not, are also set for each system.

This approach is very useful and reliable, especially thanks to the use of ML, which makes it possible to learn the normal behaviour of a system by tracking it under normal conditions. However, anomaly detection can sometimes lead to too high false positive rates. This can be due to the inner variability of some systems. Also, changes are a part of the normal behaviour of systems and thus, models must be periodically adjusted or retrained, which is not always easy if the amount of data necessary is high.

In order to cope with the above-mentioned limitations of anomaly detection techniques, the approach proposed in T5.4 of the project is based on modelling and characterizing tactical and operational intelligence, so that suspicious actions are comparable among them. In this way, APT groups can be clustered based on which tactical and operational intelligence they use when attacking a system. As a result, when an anomalous behaviour is observed and detected, it is possible to match this behaviour against each APT group cluster, assess the possibility of being under an attack carried out by one of the APT groups in these clusters and raise an alert.

Also, it is possible to alert cyber-security analysts about other actions usually associated with these APT groups so that they can look for any of these actions in case they no one noticed before or to be prepared for next stages of the attack.

In addition to all the above mentioned, once an APT is detected, it is important to be able to determine which APT group carried out the attack. Attribution is important, not only to be able to prosecute cybercriminals, but also as a source of Threat Intelligence. Knowing the APT group behind an APT attack and understanding their final motivations can be useful for identifying actions which may have not been previously detected, for identifying other potential targets in the organization, industry sector, country, etc.

The interdependencies within the grid, the highly complex environment which introduces an expanded attack surface introduce major challenges that will be addressed by R^2D^2, developing the Supply Chain Assessment toolkit – that contains the Sandbox tool and the Self-assessment tool.

CYBER, having identified the complexity of the Supply Chain in a complex and critical environment such as the EPES, develops a new tool, called The Sandbox tool, capable of overviewing and classifying the operations of a newly acquired software or a software update in an isolated environment. This tool not only integrates the capabilities of the partner`s solutions (CARMEN tool, Claudia tool, List, Argos, KSI tool) but further improves the process to evaluate the security status of a new piece of software before integrating it in the production environment. In detail, the Sandbox tool introduces a unique approach to:

- have a whole sandboxed staging environment;
- have common trusted repository of evaluated SW in a trusted community;
- integrate the staging with SIEM / machine learning;
- integrate evaluated software with a closed community block chain.

Additionally, CYBER will develop a new tool, the Self-Assessment Tool, to strengthen EPES and their vendor supply chain capabilities helping them to adapt to the international standards and widely accepted methods in order to define and implement security baselines and risk management methodologies that will address the supply chain cyber-security risks. The adoption of such a tool will assist operators in the energy sector to work on comparative results to identify gaps in their security posture compared to the levels demonstrated by others in the community. In detail, the Self-assessment tool introduces a unique approach for EPES Operators and their suppliers to:

- establish a single point of reference with the Supply Chain security standards and guidelines;
- evaluate themselves against the standards and identify supply chain risks in order to deploy the required controls;
- evaluate themselves against the average scoring of the EPES community;
- assess the cyber security posture of each existing or potential supplier.

Another testing of the tokenization tool will be done in the context of Smart meter data transferring. For this, the Greek pilot will be used. Here, there is a special type of smart meter deployed with special characteristics. Basically, the smart meter features a CPU that can be used (among other things) to apply special transformation to the messages before being sent

to the AMI. ETRA will perform tokenization of the messages before being sent through MQTT. This will prevent tampering of messages linked to fraud.

3.3 RELEVANT USE CASES AND ACTORS

This section summarises 10 use cases presented in Table 3 that have been defined under WP2 by WP5 participants. In addition, Table 3 lists actors which are related to use cases and tools. In the R^2D^2 project there are two types of actors – organisational and functional (tool, component of tool or other system) actors. This section also describes 39 requirements for six tools presented in Table 4 that are developed and tested under WP5.

Table 3 – Use cases and actors

ID	Title	Task	Tool(s)	Actor(s)

7	Enhancement in DER control and management systems to participate in flexibility procurement schemes for DSO and TSO to improve network operation security	5.3	• UniFusion	• Organisational:
				  ○ DSO
				  ○ DER/RES
				  ○ SCADA/ADSM
				  ○ SCADA (RES)
				  ○ Flexibility system
				• Functional:
				  ○ Flexibility system
				  ○ UniFUsion

10	Improving of LV network observability based on billing metering system by means of secure interface with SCADA-ADMS system	5.3	• UniFusion	• Organisational:
				  ○ DSO
				  ○ Active consumer
				  ○ DER
				  ○ SCADA/ADSM
				  ○ SCADA (RES)
				• Functional:
				  ○ Flexibility system
				  ○ Communication gateway
				  ○ UniFusion

11	DSO - TSO congestion and power quality coordination in application of system services	5.3	• UniFusion	• Organisational:
				  ○ DSO
				  ○ TSO
				  ○ Aggregator
				  ○ SCADA (RES)
				• Functional:
				  ○ Flexibility system
				  ○ UniFusion

27	Monitoring communications behaviour of newly deployed components in an EPES staging environment	5.5	• Sandbox tool
			• KSI tool
			• CARMEN
				• Organisational:
				  ○ TSO
				  ○ DSO
				  ○ EPES vendors / suppliers
				• Functional

				○ Device Origin and Supply Chain Toolkit ○ Communications Monitoring System ○ Deep Learning System ○ KSI toolBlockchain
28	Adapt/Develop EPES specific vendor management & suppliers' audit practices	5.5	• Self-assessment tool	• Organisational: ○ TSO ○ DSO ○ EPES vendors / suppliers • Functional ○ Assessment Tool
33	Detection of anomalies associated with cyber-security in EPES infrastructure	5.3	• CARMEN	• Organisational: ○ CISO ○ DSO ○ TSO ○ RCC ○ System Operator ○ Intelligence Units • Functional ○ CARMEN ○ List ○ Firewall ○ Claudia ○ Argos
34	Pattern detection and correlation	5.4	• CARMEN	• Organisational: ○ CISO ○ DSO ○ TSO ○ RCC ○ System Operator ○ Intelligence Units • Functional: ○ CARMEN ○ Intelligent cyber-security Module ○ Firewall
36	Validation of network model integrity	5.1	• KSI tool	• Organisational: ○ TSO ○ RCC • Functional: ○ IGM creation tool ○ Shared repository ○ TSO Internal repository ○ RCC Internal repository ○ Merging tool (CGM creation tool) ○ KSI tool

37	Energy data tokenization	5.2	• Tokenization tool	• Organisational:
				  ○ DSO
				• Functional:
				  ○ DSO`s database
				  ○ Tokenization tool

38	DSO grid balancing data tokenization	5.2	• Tokenization tool	• Organisational:
				  ○ DSO
				• Functional:
				  ○ Balancing data creation system
				  ○ DSOs internal database
				  ○ Tokenization tool

Table 4 – Tool-related requirements

ID	Title	Tool/Task
PRE_001	Relevant changes in data which may affect ML-based components must be detectable	• CARMEN/T5.4
PRE_002	Web service for signing and verification	• KSI tool/T5.1 • Tokenization tool/T5.2 • CARMEN/T5.3, T5.4
PRE_003	Command line solution for signing and verification	• KSI tool/T5.1 • Tokenization tool/T5.2
PRE_004	Connections to KSI Gateway	• KSI tool/T5.1 • Tokenization tool/T5.2
PRE_005	Connections to KSI Blockchain	• KSI tool/T5.1 • Tokenization tool/T5.2
PRE_006	Sample data sets for tools available as soon as possible after UC-s are defined and agreed	• KSI tool/T5.1 • Tokenization tool/T5.2
PRE_007	KSI tool detects 100% of changed in signed data during verification	• KSI tool/T5.1
PRE_008	KSI tool confirms integrity of original files on 100% of cases, when verification function is applied on originally signed data	• KSI tool/T5.1
PRE_009	Tokenization tool provides tokens, which verify originality of tokenized data in 100% of cases	• Tokenization tool/T5.2
PRE_010	Model (IGM, CGM) shall be available	• KSI tool/T5.1
PRE_011	Data format must be agreed to run data registration and integrity validation	• KSI tool/T5.1 • Tokenization tool/T5.2
PRE_012	Tokenized data and its token must be stored in the same or different database, a link between them must be maintained	• Tokenization tool/T5.2
PRE_013	PRECOG product must be able to comprehend and analyze ICS protocols	• CARMEN/T5.3, T5.4
PRE_014	PRECOG Supply Chain Assessment Tool must provide management guidelines for EPES to secure supply chain	• Supply Chain Assessment Tool (Self-assessment tool)

PRE_015	PRECOG Supply Chain Assessment Tool must provide guidelines for EPES' vendors to use to secure their supply chain and their product development	• Supply Chain Assessment Tool (Self-assessment tool)
PRE_016	PRECOG Supply Chain Assessment Tool must provide guidelines in HTML and PDF format	• Supply Chain Assessment Tool (Self-assessment tool)
PRE_017	PRECOG Supply Chain Assessment Tool shall support a self assessment for EPES Operator vendors/suppliers to evaluate their current supply chain and development practices	• Supply Chain Assessment Tool (Self-assessment tool)
PRE_018	PRECOG Supply Chain Assessment Tool shall support a self assessment for EPES Operator to evaluate their own supply chain management practices	• Supply Chain Assessment Tool (Self-assessment tool)
PRE_019	PRECOG Supply Chain Assessment Tool should provide scoring mechanisms to assess and evaluate vendor and EPES practices, providing an overall rating or score	• Supply Chain Assessment Tool (Self-assessment tool)
PRE_020	PRECOG Supply Chain Assessment Tool should be accessible through standard web browsers and compatible with different devices, such as desktops, tablets, and smartphones	• Supply Chain Assessment Tool (Self-assessment tool)
PRE_021	PRECOG Supply Chain Assessment Tool shall be accessible to registered/authorised users to enforce proper access control on the provided information and the assessment results	• Supply Chain Assessment Tool (Self-assessment tool)
PRE_022	PRECOG Supply Chain Assessment Tool should generate comprehensive reports summarizing the assessment results for vendors and EPES practices	• Supply Chain Assessment Tool (Self-assessment tool)
PRE_023	Supply Chain Assessment Toolkit must get access to T5.1 tool to register and validate data and its associated proofs	• Supply Chain Assessment Tool (Sandbox tool)
PRE_024	PRECOG product will automatically detect new devices connected to the network	• CARMEN/T5.3
PRE_025	PRECOG will detect potential threats using pattern detection	• CARMEN/T5.3, T5.4
PRE_026	PRECOG product will detect anomalies based on traffic characterization	• CARMEN/T5.3
PRE_027	PRECOG product will detect anomalies based on control, operation and supervision levels	• CARMEN/T5.3
PRE_028	PRECOG will detect operational alerts from IT/OT devices	• CARMEN/T5.3, T5.4
PRE_029	PRECOG Supply Chain Assessment Tool should monitor new components communications in an isolated (staging/test) environment and for a specific period of time, to identify suspicious communications	• Supply Chain Assessment Tool (Sandbox tool)
PRE_030	PRECOG Supply Chain Assessment Toolkit should identify suspicious communications utilizing T5.4 Deep Learning Data Analytics Module	• Supply Chain Assessment Tool (Sandbox tool)
PRE_031	PRECOG Supply Chain Assessment Toolkit shall utilise the R^2D^2 blockchain to protect the integrity of the assessment results	• Supply Chain Assessment Tool (Sandbox tool)

PRE_032	PRECOG Supply Chain Assessment Toolkit should maintain a list of evaluated components on the Device Origin and Supply Chain Toolkit available to the EPES community	• Supply Chain Assessment Tool (Sandbox tool)
PRE_033	PRECOG should assure secure communication between DSO "Flexibility system" and devices	• UniFusion platform/T5.3
PRE_034	PRECOG should alarm DSO IT security department in case of detected attack	• UniFusion platform/T5.3
PRE_035	Validation environment for should be offline	• KSI tool/T5.1
PRE_036	Environment will be established using virtual machines	• KSI tool/T5.1
PRE_037	Claudia tool has to be installed in a windows device to detect anomalies to act as EDR	• Claudia tool/T5.3
PRE_038	Claudia tool needs dependencies for installation	• Claudia tool/T5.3
PRE_039	CARMEN needs internet connection to S2 servers	• CARMEN/T5.4

3.4 APPROACH

This subchapter describes approach used to conduct work in WP5. All WP5 tools developed under tasks T5.1 – T5.5 follow Smart Grid Architecture Model (SGAM) architecture methodology defined in D2.1 (Requirements and Detailed Architecture Design). Table 5 (below) presents eight steps that were mandatory for all WP5 participants to follow to deliver the six PRECOG tools.

Step one: The activities started under WP2 (Project Foundations) where all R^2D^2 participants worked with potential use case ideas. It was practically also the start for WP5 as it helped to set up participants and plan the workflow. WP5 description from proposal and experience of tool providers and pilots were used as input in this phase. Expert analysis with text content analysis were conducted and expert discussions were held over potential use cases. Use case definition forms (delivered in WP2) were part of the approach used to deliver a pool of use cases.

Step two: As a part of WP2, WP5 participants used the pool of UCs as input to create selection of UCs with mutual confirmation from pilots and technology providers that selected UCs are input for further work, which leads to practical solutions. Three approaches were used – use case revision forms, expert analysis combined with text content analysis and expert discussions. From the pool of potential use cases selection of use cases was done.

Step three: In this phase UCs were used as input. UCs and related tools were analysed by pilot representatives and tool providers in the format of workshops, expert discussions and text content analysis. Volere approach (Volare tool introduced in WP2) was used in the format of the Volere tool provided by ETRA for gathering and analysing information. As a result, tool-related requirements were formulated.

Step four: Here the selection of UCs and tool-related requirements was used as input. The SGAM was used as the mainframe for R^2D^2 architecture. Expert analysis combined with text content analysis and expert discussions were used as an approach. In addition, one workshop was conducted in the format of a physical meeting. This resulted in architecture for tools.

Step five: In this phase the selection of UCs with new data collected in previous steps was fine-tuned to increase detail of use cases and understanding between pilot partners and technology providers. The selection of UCs, tool-related requirements and architecture for

tools were used as input information for this step. Through expert analysis, text content analysis and expert discussions two outputs were achieved: the final list of UCs and the definitions and tool descriptions.

Step six: The finalised list of UCs and definitions and tools descriptions was used as input in this work phase. The KPI templates were used to gather and organise information. Expert analysis, text content analysis and expert discussions were used as approaches to work with input data. The KPIs for tools were delivered to ensure specific objectives SO1 and SO3 will be reached by tools provided in WP5.

Step seven: This step prepares the content for T6.5 (Integration and UI) that starts after the D5.1 delivery. The finalised list of UCs and definitions combined with tools' descriptions were used as input. Expert analysis, text content analysis and expert discussions were used as approaches to analyse and compile the material. As a result, requirements for user interface were provided.

Step eight: This phase finalised the conducted work and delivers a deployment plan as input for D5.2. Step eight used the following four inputs: finalised list of UC`s and definitions, internal architecture for tools, tool-related requirements and user interface requirements. Expert analysis combined with text content analysis were conducted on input data. In addition, expert discussions and quantitative analysis were used as an approach to get results from input data.

Table 5 – Approach used to organise work in WP5

Step nr.	Step	Input	Approach	Output
1	UC definition	• WP5 description • Experience of tool providers and pilots	• Use case definition forms • Expert analysis combined with text content analysis • Expert discussion	• Pool of UC-s and definitions
2	UC review	• Pool of UCs and definitions	• Use case revision forms • Expert analysis combined with text content analysis • Expert discussion	• Selection of UCs and definitions
3	Requirements mapping	• Selection of UCs and definitions	• VOLERE approach and Volere tool • Expert analysis combined with text content analysis • Expert discussion • Workshop	• Tool-related requirements

			• Quantitative analysis	
4	architecture for tools	• Selection of UCs and definitions • Tool-related requirements	• Smart Grid Architecture Model (SGAM) • Expert analysis combined with text content analysis • Expert discussion • Workshop	• architecture for tools
5	UCs fine tuning	• Selection of UCs and definitions • Tool-related requirements • Internal architecture for tools	• Expert analysis combined with text content analysis • Expert discussion	• Finalised list of UC-s and definitions • Tools descriptions
6	KPI-s mapping for tools	• Finalised list of UC-s and definitions • Tools descriptions	• KPI templates • Expert analysis combined with text content analysis • Expert discussion • Quantitative analysis	• KPIs for tools
7	User Interface requirements	• Finalised list of UC-s and definitions • Tools descriptions	• Expert analysis combined with text content analysis • Expert discussion	• User interface requirements
8	Deployment planning	• Finalised list of UC-s and definitions • architecture for tools • Tool-related requirements • UI requirements	• Expert analysis combined with text content analysis • Expert discussion • Quantitative analysis	• Deployment plan (design) for tools

4 Product Description

This document presents the first iteration in the R^2D^2 project, where tools and developments are described and deployment plan is delivered. Figure 1 illustrates the tools in PRECOG, highlighting the relation of tools, tool providers, tasks and pilots (Greek, Serbian and Slovenian pilots). WP5 aforementioned list of tools is as follows:

- **The KSI tool** (T5.1), to be tested in UC36 (pilot partners SCC and ELEK) and UC27, developed by GUARD.
- **The Tokenization tool** (T5.2) to be tested in UC37 (pilot partner HEDNO) and UC38 (pilot partner ELEK), developed by GUARD.
- **The CARMEN tool** (T5.3 and T5.4) to be tested in UC33 (pilot partners SCC, EMSS, HEDNO and ELEK) and to be tested in UC34 (pilot partners SCC, EMSS, HEDNO and ELEK), developed by S2.
- **The UniFusion platform** (T5.3) to be tested in UC7, UC10 and UC11 (all use cases with ELEK and ELOVE), developed by ELPROS.
- **The Sandbox tool** (T5.5) to be tested in UC27 (pilot partners EMSS, HEDNO and ELEK), developed by CYBER.
- **The Self-assessment tool** to be tested in UC28 (T5.5) is compiled by CYBER (pilot partner HEDNO).

As it will be detailed later in this document, CARMEN is going to be deployed in collaboration with the ELEK/ELPROS pilot using the UniFusion platform developed by ELPROS. Thus, collaboration and synergies will take place between use cases UC7, UC10, UC11, UC33 and UC34. In addition, CARMEN will also be deployed in the Serbian pilot using the Sandbox tool developed in T5.5 and, as a result, synergies and collaboration will also take place.

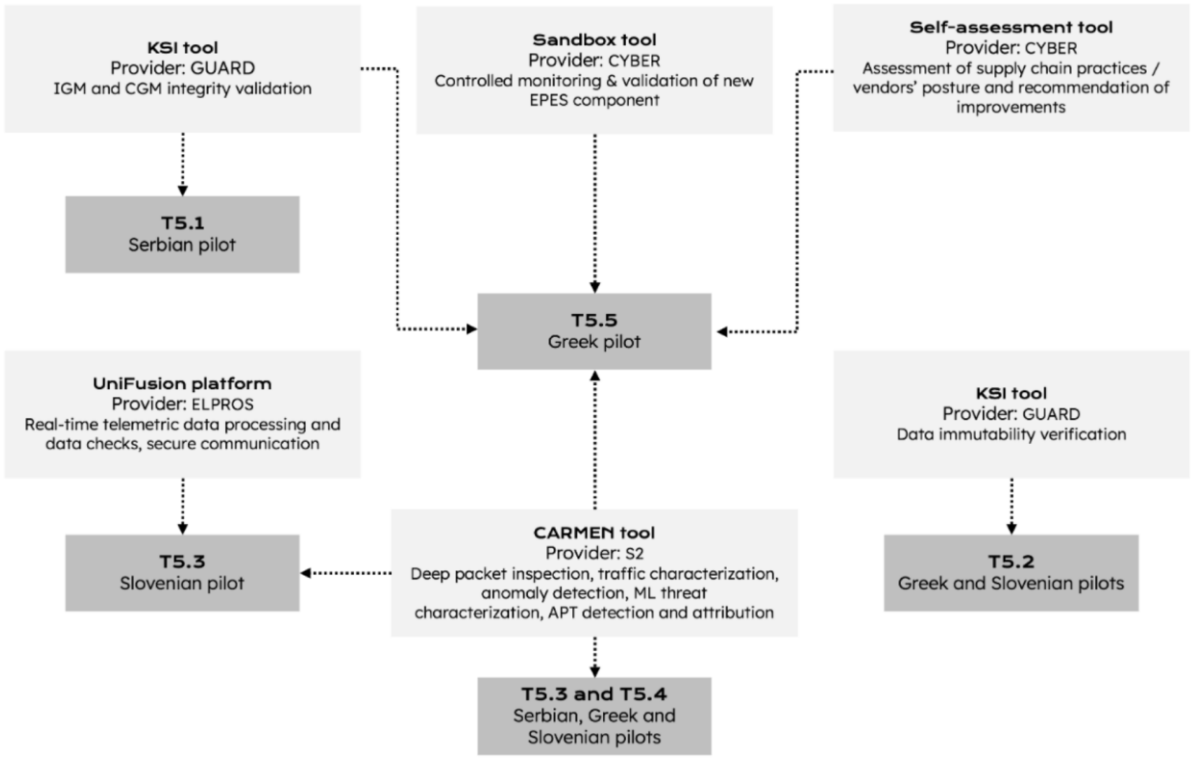

Figure 1 – PRECOG tools and related pilots

GUARD provides data integrity validation through the KSI tool and data immutability verification through the Tokenization tool. KSI tool is tested in collaboration with the Serbian partners RCC and EMSS where data integrity validation is provided to grid models. Tokenization tool is tested with the Greek pilot HEDNO and Slovenian pilot ELEK. With HEDNO the energy consumption and with ELEK the grid balancing data is tokenized. KSI tool is developed and enhanced to be suitable for the automatic data integrity validation in communication between different parties who could rely not only on the data integrity validation upon receiving the data but also use it for auditing and validation for decades to come. Tokenization tool is enhanced to enable data tokenization on a granular level, such as measurement data entries in a periodic report (that could range from minutes to daily periods), and enabling data tokenization on the IoT devices, such as smart meter itself, to eliminate data tampering possibilities from source to data storage.

GUARD's aim of the project is to successfully implement aforementioned functionalities and achieve technical KPI-s, so pilots can 100% trust their data. In addition, collaboration with Serbian, Greek and Slovenian pilots will produce theoretical and practical experience which has potential for business activities beyond the R^2D^2 project.

The main tool that S2 Grupo will use for covering use cases UC33 and UC34 is the CARMEN tool [7]. CARMEN is S2 Grupo's APT (Advanced Persistent Threat) compromise detection tool, able to detect anomalies and misuse (improper or illegal use of any computer in a system).

Within the scope of the project, CARMEN's capabilities will be enriched in several aspects:

- Development of new dissectors to work with protocols specific to the electrical sector, such as IEC 61850 or ICCP 60870-6/TASE.2. The inclusion of these new protocols as additional inputs will let CARMEN use packet inspection capabilities of these specific

protocols to analyse the Smart Grid traffic to identify the above mentioned APT attacks (more specifically, CARMEN will raise an alert that a human analyst will validate).
- Development of preprocessing and aggregation methods for tracking, correlating, and analysing different data sources, covering both network traffic protocols and IoT devices which may be deployed in the monitored system.
- Development of AI-based capabilities for detecting and attributing APT threats based on their similarity to previously pre-processed and normalised tactical and operational intelligence.

As it will be detailed later in this document, since CARMEN is going to be deployed in the Slovenian pilot, needed data will be provided by UniFusion platform. Thus, collaboration and synergies will take place between use cases UC7, UC10, UC11, UC33 and UC34. In addition, CARMEN will also be deployed in the Slovenian pilot using the Sandbox tool developed in T5.5 and, as a result, synergies and collaboration will also take place.

ELPROS platform UniFusion will be upgraded with additional functionalities which are required in use cases (UC7, UC10 and UC11). This will include:

- Implementation of procedures which fulfil requirements by the latest standard IEC 62351-3 for communication network and system security in communication protocols IEC 60870-5-104, ICCP/TASE.2 and MQTT.
- Implementation of access filtering from allowed IPs.
- Implementation of multi-level login procedures.
- Automatic data checks that can detect attempts of unauthorized data read, receiving data outside the expected limits, communication requests from unapproved addresses.

Supply chain cyber risk is becoming an evolving area of potential vulnerabilities and threats that can arise in the interconnected network of suppliers, vendors, manufacturers, and service providers. In today's highly digital and globalized business environment, the complexity of involving partners across various regions and industries is increasing. While this approach offers numerous advantages in terms of efficiency and cost-effectiveness, it also opens new paths for cyber threats and attacks. Key aspects of supply chain cyber risk include:

- Interconnections: Supply chains involve a web of interconnected entities, each with its own cyber-security protocols and measures. Any weak link in this chain can expose the entire web to potential cyber threats.
- Third-party vulnerabilities: Organizations rely on third-party vendors and suppliers to provide products and services. If any of these external entities lack robust cyber-security practices, they can become entry points for attackers to compromise the entire supply chain to:
    - Gain unauthorized access to sensitive data, intellectual property, customer information, or financial data. Such breaches can lead to reputational damage, regulatory penalties, and financial losses.
    - Execute malware and ransomware attacks, disrupting operations of entire supply chain's organization by encrypting critical data, demand ransom payments.
- Counterfeit products: Cybercriminals might tamper with products during the building and distribution process, leading to safety risks for end consumers.
- Lack of visibility and control: Due to limited visibility into the cyber-security practices of their suppliers and partners, it is challenging to assess and manage potential risks effectively.
- Delays in response and recovery: When a cyber-attack occurs within the supply chain, it requires additional time to identify the breach and coordinate responses with multiple stakeholders can significantly impact the overall recovery process.

The importance of supply chain cyber-security has become crucial for organizations to secure their operations, protect their customers' data, and maintain the integrity of their products and services in an interconnected world.

Supply chain cyber risk management requires a comprehensive approach that includes:

- Assessing the cyber-security practices of all supply chain partners and vendors.
- Establishing cyber-security standards and requirements for all entities within the chain.
- Regular monitoring and auditing the security practices of suppliers and partners.
- Implementing robust encryption, access controls, and network security measures.
- Creating incident response and recovery plans that involve all relevant stakeholders.
- Enhancing collaboration and information sharing among supply chain partners to detect and respond to threats more effectively.
- Monitoring the new products operation before rolling them out in the production environment.

To support the requirements of risk management of the supply chain the “T5.5 - Device Origin and Supply Chain Toolkit” will introduce a framework for the EPES community. This framework is consisted of:

- A set of industry best practices and standards for the EPES regarding supply chain security management and supplier’s auditing
- A set of industry best practices and standards for the vendors regarding supply chain security management and software development security standards
- Self-assessment questionnaires for the EPES to monitor the security posture of their supply chain.
- Self-assessment questionnaires for the vendors to monitor the security posture of their supply chain.
- Benchmarks regarding their security posture against the community and/or the standard.
- The capability to monitor new components communications, evaluating and classifying the results before their deployment in the production environment. With this tool and process, security teams of all EPES will be able to identify and track potential cyber threats related to supply chain attacks and take appropriate measures to mitigate them without endangering their infrastructure. Monitoring can be used to detect any malicious activities, or even misconfigurations, which can then be addressed before they can be exploited by attackers.

4.1 TASK 5.1 – IDENTIFICATION AND AUTHENTICATION OF ENERGY IOT AND EDGE DEVICES

In T5.1 GUARD is testing and developing the KSI tool in UC36 based on Serbian infrastructure. Two partners from the Serbian pilot, SCC and EMSS are involved in T5.1. The goal of these partners is to improve trust in network models (defined in UC36 definition), which are created and used in many phases by different users.

4.1.1 KSI tool

Internal Architecture of the tool

The aim of the KSI tool in Serbian pilot is to deliver integrity validation functionality, which provides 100% trust for the data signed with the KSI tool. The KSI tool is developed for UC36 (Validation of network model integrity), which can be shortly described as follows: a) TSO creates IGM; b) stores it; c) sends it to RCC; d) RCC uses IGM as input and creates CGM from this; e) then RCC shares CGM with other RCCs and TSOs. In every step (starting with IGM creation) it is crucial to have 100% trust for data integrity as using poisoned/corrupted data may cause loss in assets and cause power outages. Figure 2 illustrates the component layer diagram of UC36 described in D2.1 (Requirements and Detailed Architecture Design) [3].

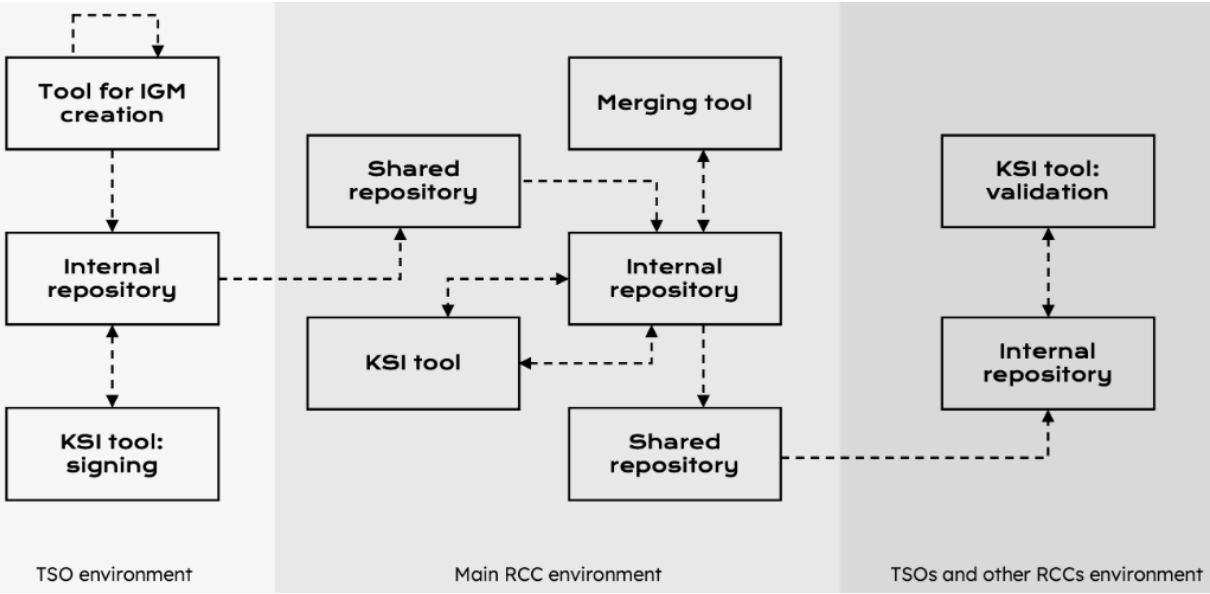

Figure 2 – UC36 component layer diagram

The KSI tool is composed of two components running on the pilot's system, providing data signing and integrity validation functionalities. The signing component relies on the blockchain network to generate the signatures for later validation. The tool would take incoming requests that are either already hash digests or raw data itself that then would be hashed based on the chosen algorithm. The resulting hash digest and additional hash entries and metadata is then aggregated into a root, by hash using Merkle tree (where each data point hash is a leaf), that then is sent to the blockchain network for registration in the next block. Once a new block is committed to the blockchain, the network returns a proof. Tool then generates signatures for each data point using the initially built Merkle tree and proof received from blockchain. Each generated signature integrity is validated and finally corresponding signatures are used to respond back to each specific data registration request. In case of validation (validation component) the tool expects the data (or its hash digest) and signature as input. Tool will hash the data to get the digest and proceeds to compare the hash digest with the signature, if this fails then it will respond back with an “Invalid document” message meaning that provided data (or hash digest) and signature do not match. If the comparison succeeds, then the signature itself is internally validated for any inconsistencies using hash based cryptography to validate the integrity and registration time and finally the root of the signature is compared against the blockchain. In case of any inconsistencies during validation, the tool responds back with a corresponding code and message to indicate the detected issue and to enable poisoned or corrupted files to be filtered out for further investigation.

The KSI tool is used via HTTP API using JSON data format and is intended for developers or advanced users. As in the majority of use cases, the KSI tool runs in the background of other systems and services.

In UC36 TSO creates IGM, which is then signed with KSI tool and later used as input for CGM creation after IGM has been validated on the RCC side, where CGM as the final product is also signed with KSI tool and shared to a third party. KSI tool does not have requirements for data size or format, nor granularity of the signing process. When implementing the KSI tool to the data processes it is important to provide the data for signing and validation the same way, having the same pre-processes if applicable as even the smallest changes (extra or missing white space) would lead to validation failure. When using the KSI tool for data signing and integrity validation outside UC36, the same principles apply. In UC27 (Monitor communications behaviour of newly deployed components) CYBER is combining KSI tool with CARMEN tool and Sandbox tool. In that use case, KSI tool provides the same functionality as in UC36 – it is used to sign data and provides integrity validation for signed data. Subchapters 4.3.1 and 4.5.1 describe in more detail how the KSI tool is combined with the Sandbox tool and CARMEN tool in use cases UC33 and UC34.

No UI is developed for the use case. However, if the pilot use case should evolve into a direction where UI could become useful, then a simple UI could be designed to enable manually sign and validate data and its signature.

Resources

The KSI tool will be developed in java and is run as a service that is built using Spring Boot with either Gradle or Apache Maven. Service is used via HTTP API using JSON data format. Service is packaged and run in docker container for ease of operation, alternatively it could be run natively if need be. The tool itself is internally using standard java libraries and classes, such as MessageDigest for data hashing, and BouncyCastle to complement the default Java Cryptographic Extension (JCE). KSI tool communication with the blockchain network is using type-length-value (TLV) messages over HTTP that only contain hash digests.

For blockchain technology choice wise it is currently envisioned to use permissioned blockchain to avoid any additional resource requirements on infrastructure whilst keeping the blockchain database distributed and have high performance. Blockchain components in codebase are included as specific implementations to allow its replacement in the future if there is a need or desire for a shift to different permissioned blockchain, towards private or even public blockchain.

4.2 TASK 5.2 – ENERGY TOKENS AND TRADING CERTIFICATES SECURITY

In T5.2 GUARD is developing and testing the Tokenization tool in two use cases in collaboration with two pilots. The UC37 (Energy data tokenization) tool is developed with the Greek pilot HEDNO and in UC38 (DSO grid balancing data tokenization) the Tokenization tool is developed with Slovenian pilot ELEK. The pilots aim to test innovative systems inT5.2 that can increase trust of the data crucial for both in business and operative layers in the pilot`s infrastructures. HEDNO needs to be sure that archived consumption data can be trusted and for ELEK it is important that the grid balancing data is fully intact and can be applied on substations and RTUs.

4.2.1 Tokenization tool

The aim of the Tokenization tool in Greek pilot is to provide immutability verification functionality, which provides 100% trust for the data that has been tokenized with the Tokenization tool. Data to be secured for the network operator involves SCADA system, as well as AMI from telemetered customers. The basis for tool development is UC37 which, in short description, follows the next steps. The data is collected from smart meters, then stored in a database and tokenized. Tokens are later used for data verification that is configured to occur periodically or on demand. Figure 3 illustrates the architecture of UC37 described in D2.1. Tokenization tools aim in UC38 is to provide authenticity proofs for grid balancing messages that DSO is publishing for participants in grid operation, mainly intended for the energy producers and balancing service providers. In short, DSO creates grid balancing data, then tokenizes it and stores both data and tokens in an internal database. If DSO or a third party intends to use grid balancing data, Tokenization tool is used to verify data immutability. Figure 4 illustrates the architecture of UC38 described in D2.1.

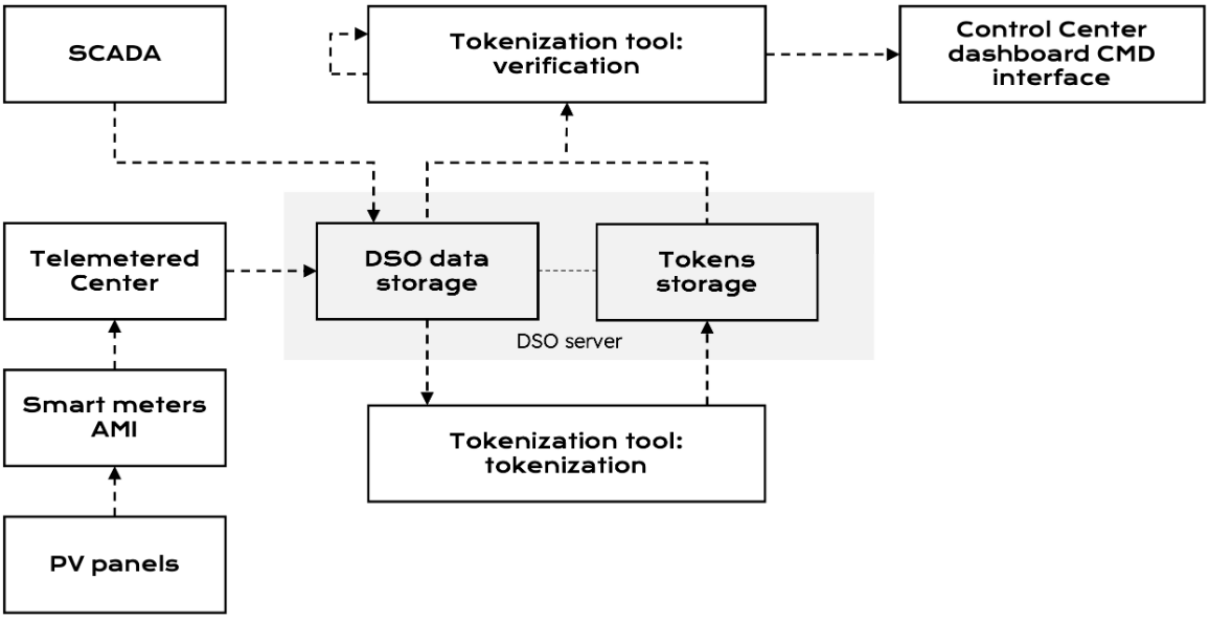

Figure 3 – UC37 component layer diagram

Internal architecture of the tool

In UC37 consumption data is automatically collected from AMI smart meters and then stored in an internal database. Tokenization tool runs in parallel with HEDNOs internal data storage (during the WP5 development replica of real life storage is used) that is used to tokenize consumption data. After the data automatic collection, data is tokenized and stored, providing the earliest possible verification point for stored data and its later usage. Tokenization tool has two main functionalities – token creation and token verification, both of which are consumed using HTTP API in JSON data format. Tokenization tool makes use of a one-way cryptographic hash function to generate tokens to substitute the potentially sensitive data into non-sensitive equivalent, making it impossible to reverse to original data as long as secure one-way hash functions are being used. Token generation (or issuance) makes use of blockchain to immutably record each generated unique token without placing any user data onto the blockchain but instead some hash digests. Tokenization tool then returns the created token to the requester. Such an approach allows tokens to be independently verifiable within and outside of the system where those were generated, and it does not reveal any sensitive

information to any of the parties nor via blockchain. Data and its tokens are to be verified before usage to detect any accidental or intentional tampering and to eliminate untrusted and compromised data from further processes. Token creation and verification is designed to be an automatic process, running in the background when implemented into an existing system. The tool, in case of any errors during token creation or verification, responds back with the corresponding message to take appropriate measures, from issue investigation to taking countermeasures to detect data inconsistencies.

In UC38, the fundamental technology of the Tokenization tool is similar to UC37, which makes development activities easier and helps to perform baseline testing activities more efficiently. Difference comes from pilots' needs and nature of their infra systems. In UC38, grid balancing data is generated based on energy production, consumption and quality. Grid balancing data is then tokenized and tokens are stored in the database for further immutability verification.

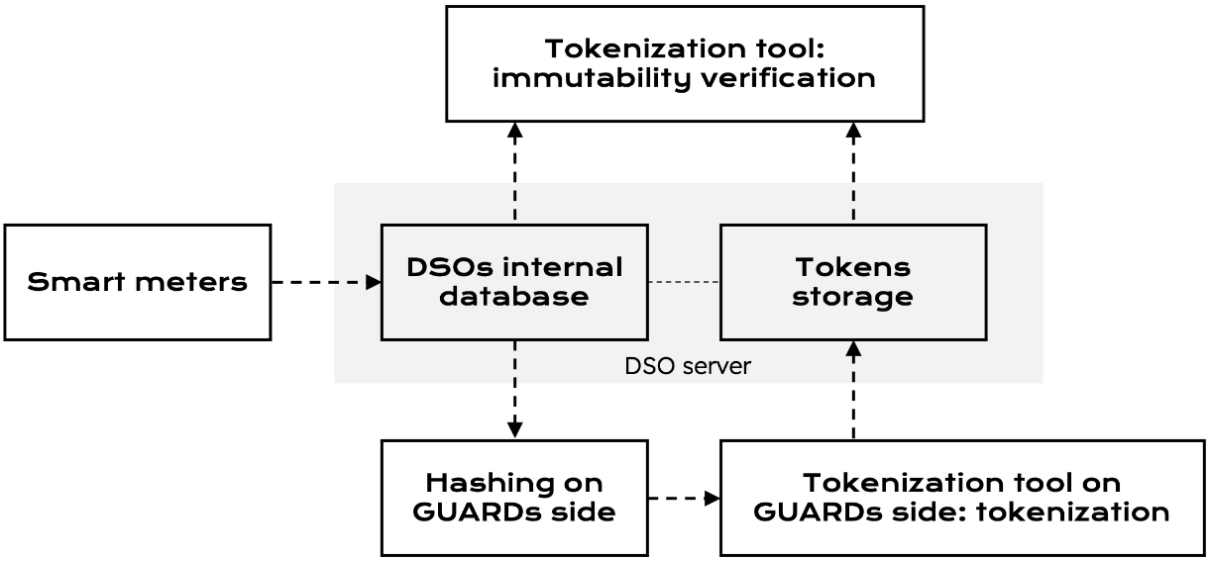

Figure 4 – UC38 component layer diagram

The tokenization tool runs in parallel with ELEKS`s internal services that are used to manage and store grid balancing data and corresponding tokens. Tokenization tool has two main functionalities – token creation and token verification both of which are consumed using HTTP API in JSON data format. The tool makes use of a one-way cryptographic hash function to generate tokens to substitute the potentially sensitive data into non-sensitive equivalent making it impossible to reverse to original data as long as secure one-way hash functions are being used. Token generation (or issuance) makes use of blockchain to immutably record each generated unique token without placing any user data onto the blockchain but instead some hash digests. Tokenization tool then returns the created token to the requester. Such an approach allows tokens to be independently verifiable within and outside of the system where those were generated and it does not reveal any sensitive information to any of the parties nor via blockchain. Data and its tokens are to be verified before usage to detect any accidental or intentional tampering and to eliminate untrusted and compromised data from processes. Token creation and verification is designed to be an automatic process, running in the background and when implemented into an existing system. The tool, in case of any errors during token creation or verification, responds back with the corresponding message to take appropriate measures, from issue investigation to taking countermeasures to detect grid balancing data inconsistencies.

The tokenization tool communication with other use case components is envisioned to be using HTTP API endpoint for serving requests. Communication is using JSON data format for its ease of use and widespread support. The same applies for tokenization tool communication with blockchain unless another service, that doesn't use HTTP API and doesn't support JSON, turns out to be a better choice. Specific request and response formats are yet to be designed and agreed upon in more detailed discussion with the pilot to best support their needs.

No UI is developed. However, if the pilot partner's use case should evolve into a direction where UI could become useful, then a simple UI could be designed to enable manually tokenized consumption data, to verify tokenized data and to view metadata within the token.

Resources in UC37 and UC38

Tokenization tool is most likely developed in Java and built using Spring Boot with either Gradle or Apache Maven. Service is consumed via HTTP API using JSON data formats. Service is packaged and run in docker container for ease of operation, alternatively it could be run natively if need be. Tokenization tool communication with blockchain networks is using JSON based data formats over HTTP API for transaction requests (recording of token on blockchain, and data requests for token verification), unless another blockchain, that doesn't use HTTP API and doesn't support JSON, turns out to be a better choice.

For blockchain technology choice wise it is currently envisioned to use permissioned blockchain to avoid any additional resource requirements on infrastructure whilst keeping the blockchain database distributed and have high performance. Blockchain components in codebase are included as specific implementations to allow its replacement in the future if there is a need or desire for a shift to different permissioned blockchain, towards private or even public blockchain.

4.3 TASK 5.3 – CYBER-SECURITY EVENTS MANAGEMENT TOOLS

In T5.3 S2 Grupo is improving and developing new capabilities for the CARMEN tool. As it will be explained later in this document, these new capabilities will be validated mainly in use case UC33 (see Figure 5). The enhanced version of CARMEN, with developments made in the scope of task T5.3 (and also in T5.4) will be deployed in the Serbian pilot. In addition to that, CARMEN is also going to be deployed in the Slovenian pilot using the UniFusion platform developed by ELPROS. As a result of this deployment, collaboration and synergies will take place between use case UC33 and use cases UC7, UC10 and UC11. Finally, CARMEN will also be deployed in the Serbian pilot using the Sandbox tool developed in T5.5 and, as a result, synergies and collaboration will also take place. Finally, since CARMEN is also going to be enhanced in T5.4, which will be tested in use case UC34, synergies between UC33 and UC34 are also expected.

Developments in the scope of task T5.3 will provide CARMEN with new capabilities for ingesting and analysing data from different data sources, searching for potential threats. New detection rules will be developed and new anomaly detection agents, based on ML, will be developed and trained to detect known and unknown threats combining the analysis of data coming from the new sources incorporated in the scope of this task and other, previously existing data sources. For validation, the enhanced version of CARMEN will be connected on SCC based on Serbian infrastructure and will analyse its traffic in a passive, non-intrusive way, without interfering with the normal functioning of the system. Developments of this task will also be validated in the Greek pilot by installing the improved version of CARMEN in the pre-production environment built by HEDNO, EMSS and CYBER. In addition to that, results of this

task will contribute to use case UC10 in the Slovenian pilot, in collaboration with ELEK and ELPROS.

The UniFusion platform will be used for use cases UC7, UC10 and UC11, related to the Slovenian pilot.

4.3.1 CARMEN tool

Internal architecture of the tool in task T5.3

As explained earlier in this document, CARMEN is S2 Grupo's APT compromise detection tool, able to detect different anomalies and misuse at different levels. Within the scope of this task, S2 Grupo will enrich CARMEN data ingestion capabilities by developing new dissectors that will work with protocols specific to the electrical sector, such as IEC 61850 or ICCP 60870-6/TASE.2. Also, new capabilities will be developed to preprocess, aggregate and extract new, combined variables and also new features from both these new data sources and from already supported ones. As a result of this preprocessing and aggregation, the amount of information to be analysed and stored when searching for threats in the Smart Grid traffic, as well as the natural variability of this information, will be reduced in such a way that it will be feasible to process it effectively.

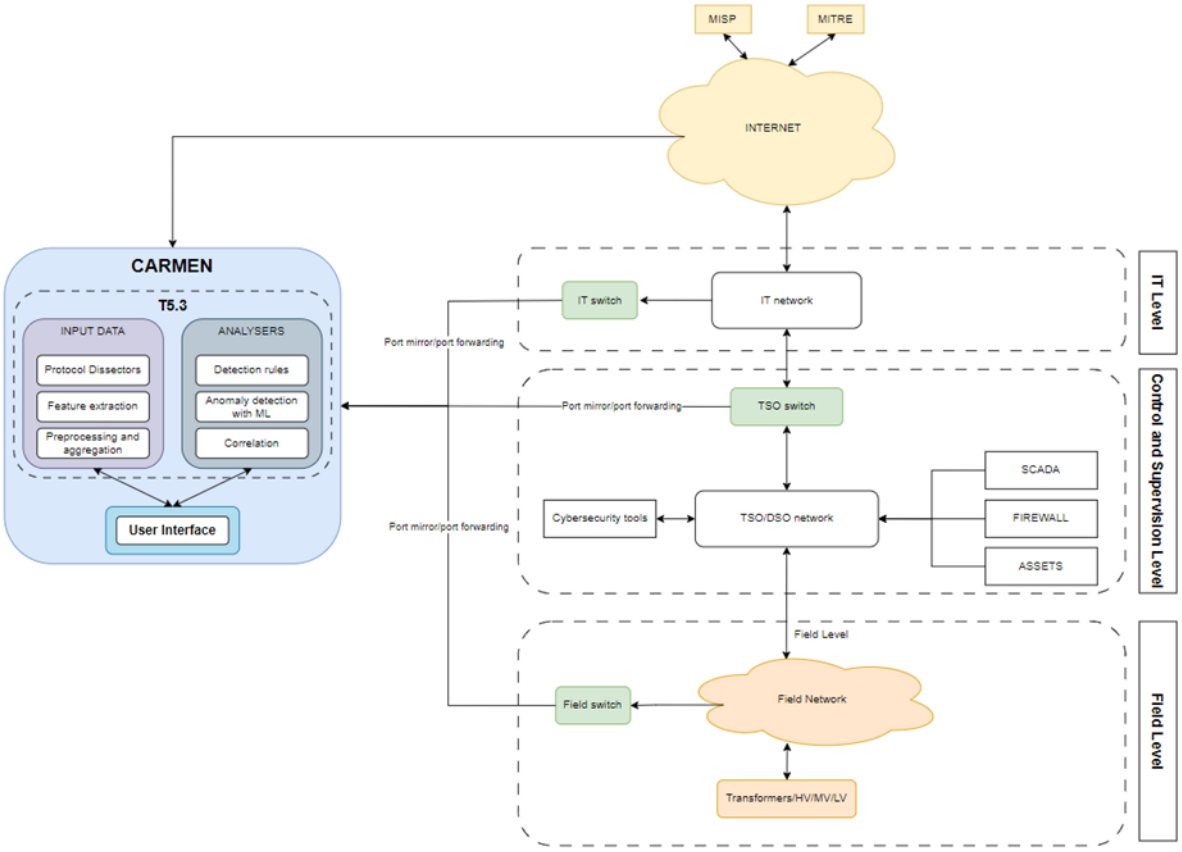

Figure 5 – UC33 component layer diagram

Thus, it will be possible to use these new data sources to develop new detection rules for already known threats and it will also be possible to train different machine learning algorithms for anomaly detection which will be used for identifying potential unknown threats, such as zero-day threats or APT threats. Every time a potential, known or unknown threat is detected, CARMEN will raise an alert that a human analyst will have to validate whenever an attack is detected.

As it can be observed in Figure 5, once deployed in a real environment, the system's network traffic will be acquired by CARMEN using port mirroring, which establishes a one-way communication channel with the pilot. In this way, the deployment of CARMEN does not generate any traffic in the network of the final system and does not interfere with its normal functioning or the quality of the service it provides.

The figure shows three possible points of connection:

- **IT Network Level**: If connected to the IT network, CARMEN will only be able to analyse the IT traffic of the final system. This traffic usually includes SQL servers, firewalls and IDS, network switches, routers, worker computers in active directory systems and other common findings. Despite network traffic hasn't specific traffic from substations, attackers could also penetrate these systems using phishing, backdoors, or similar malware for the initial attack phase, so it is important to monitor and analyse this kind of traffic too.

- **Control and Supervision Level**: If connected to the Control and Supervision network, CARMEN will be able to analyse network traffic related to SCADA system. In this network we may find SCADAS, HMI, historic log servers from substations and SCADA system, engineering stations, switches, maybe some routers, SQL servers, cybersecurity assets such as firewalls, IDS or IPS. In control network also we may find RTU, PLC or IEDs. Most of these devices use specific protocols that we should comprehend and monitor properly with the development of specific modules for CARMEN.

- **Field Level**: If connected to field level, CARMEN will be only able to analyse network traffic related to TCP/UDP. If devices use serial protocols or specific communications, CARMEN won't be able to monitor these assets. Connecting to this level without interfering with the system is not always possible due to a possible impact in the operation. That's why CARMEN only uses passive network traffic redirection from switches or other devices connected, to avoid an intrusion to field and operational networks. In this network we may find transformers, physical switches, relays, radio transceivers or surveillance cameras.

Once connected to the system and acquiring its network traffic, CARMEN will analyse the different network flows and specific industrial protocols, preprocessing, aggregating and normalizing all those data so that they can be fed to the different analysis and detection agents running in the deployed instance.

This analysis will be carried out combining different techniques. On the one hand, traditional cyber-security tools and techniques, like pattern/signature detection rules, which are able to cope with already known and characterized threats. On the other hand, different ML techniques and algorithms, such as neural networks, clustering or anomaly detection algorithms, will be trained and adjusted to learn the normal way of functioning of the system in which CARMEN is deployed and raise an alert in case of an unexpected behaviour is observed. These ML techniques are commonly used to detect unknown threats, such as zero-day threats or APT

threats. Alerts raised by each detection agent are correlated to prioritize them and to reduce the number of false positives (false alarms), which may difficult the labour of cyber-security expert analysts.

As it will be explained later in this section, CARMEN tool already has a UI which analysts can use to configure the tool, access to alerts, carry out threat hunting tasks, etc. However, in addition to this interface, generated alerts can be sent to other cyber-security tools. In the context of this project, alerts raised by the enhanced version of CARMEN deployed in the different pilots will be sent to the tool UI.

Main components developed or enhanced in T5.3

In the scope of the R^2D^2, CARMEN capabilities will be increased to cover the project use cases with the development of:

- New traffic dissectors for the above-mentioned network protocols, specific for the Power Grid. Traffic dissections are specific modules developed for specific EPES protocols that are widely used in the networks. These dissectors use deep packet inspection (DPI) based on analysing every single packet from that protocol and identifying specific command operations, registers, data, payloads, MACS, IP and much more. Some of the specific industrial protocols S2 is aiming to develop are DNP3, ICCP 60870-6/TASE.2, IEC 104 or MQTT. These protocols are EPES specific for infrastructure monitoring but there are also other that can be find in operation and supervision levels such as MODBUS, S7Comm, Profinet, or Ethernet IP. Dissectors will be developed according to surveys and demo site partners priority. This means that, at the end of the project, all demo site partner protocols that collaborate with us in UC33 and UC34 will be covered.

- Agents for analysing the different data searching for already known or potential unknown threats. These agents may rely on traditional cyber-security techniques, such as active discovering, passive vulnerability scanning or signature-based detection rules, but they may also rely in modern ML algorithms which are trained to learn the normal behaviour of the system so that they can raise an alert when an anomaly is observed. CARMEN uses other IT typical protocols such as HTTP, DNS or ICMP to monitor the system and its environment.

- Correlation agents to filter, prioritize and give context to alerts raised by the different analysis agents, so that the information which is delivered to cyber-security analysts is rich and as free of false positives as possible.

Data exchanges, communication with other tools and/or products

Regarding data exchanges, CARMEN will be connected to Sandbox tool from UC27 using port forwarding and port mirroring. The data will be collected from the preproduction environment to analyse MQTT protocol mainly. Then CARMEN may notify some alerts that may be sent by PDF or JSON to another part of the Sandbox tool to feedback the system.

Also, CARMEN will be used to report alerts from anomalous actions from the assets linked to the cyber risk assessment flow. CARMEN will be connected by port mirroring and port forwarding. After that, alerts will be sent by PDF or JSON.

Finally, ELPROS/ELEK PCAP files will be ingested by CARMEN in an offline mode and then the possible alerts will be sent to ELPROS/ELEK by PDF mainly.

The pilot's traffic will be provided to CARMEN using port mirroring. This communication is one way (CARMEN cannot generate traffic in the pilot network) and will permit Carmen to process and analyse the traffic.

The pilot's traffic will be provided to CARMEN using port mirroring. This communication is one way (CARMEN cannot generate traffic in the pilot network) and will permit CARMEN to process and analyse the traffic.

User interface

CARMEN already has a user interface that lets users see the alerts detected by the tool (see Figure 6). It is a web application so users can access via web browser to see the alerts so the analysts can work on it and make decisions based on the results provided by the tool.

User interface

CARMEN already has a user interface that lets users see the alerts detected by the tool (see Figure 6). It is a web application so users can access via web browser to see the alerts so the analysts can work on it and make decisions based on the results provided by the tool.

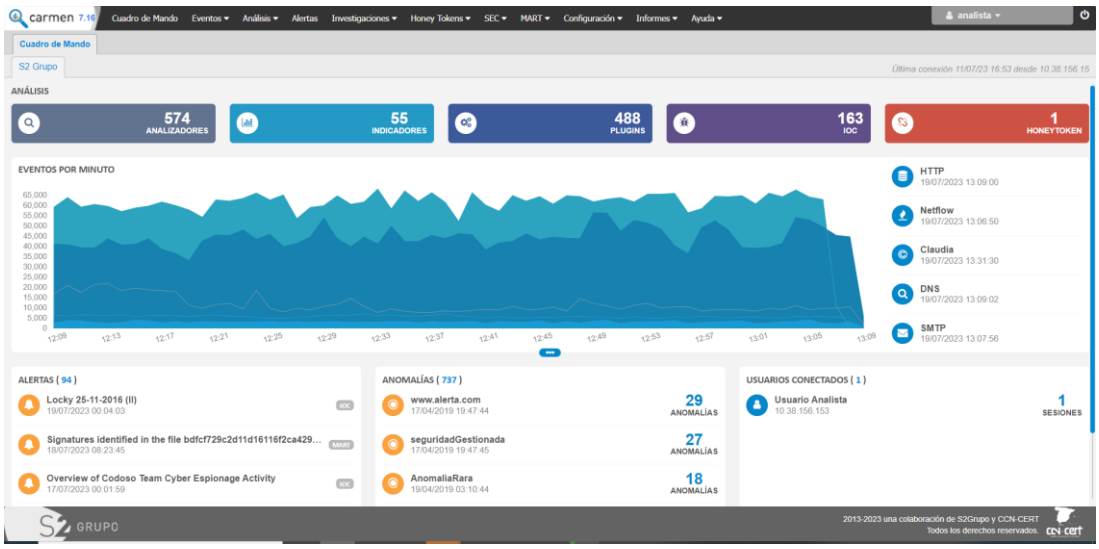

Figure 6 – CARMEN main dashboard

In addition to being able to check alerts, analysts can also see "raw" traffic information processed by CARMEN. In this case, the user interface will show different fields depending on the protocol. So, for each new supported protocol, the user interface will be enriched.

Alerts raised by CARMEN's new detection capabilities developed in the scope of the project will also be available in this UI; however, CARMEN's UI will only be available in those scenarios in which the pilot lets S2 Grupo install CARMEN. If the installation of CARMEN was not possible, then S2 Grupo would evaluate other alternatives (that could imply not having a user interface) together with each pilot.

Resources

CARMEN is set up in appliance with the following requirements for OPTIMAL installation:

- 64 GB for RAM
- 40 vCPU
- 1 disk with 300 GB for operational SAAS
- 2 disks with 2 TB SSD for log storage and retention

It is also possible to deploy a MINIMAL installation with the following requirements:

- 32 GB RAM
- 16 vCPU
- 1 disk with 300 GB for operational SAAS
- 1 disk with 1 TB SSD for log storage and retention

As it is shown in Figure 5, CARMEN requires to be connected to Internet in order to have access to MITRE [8] and MISP [9]. MITRE ATT&CK is a framework that describes the tactics and techniques employed by adversaries in cyber attacks. MISP is a platform designed to share attack information, which offers information on IoCs, tactics, tools, and others data regarding threat intelligence. MITRE and MISP are the main open source threat intelligence sources from which CARMEN acquires Indicators of Compromise (IoC), Indicators of Attack (IoA), artifacts and other accessible intelligence necessary to improve CARMEN's detection capabilities and the quality of the alerts it may raise.

Regarding the developments to be carried out in the scope of this task, these will mostly be done in Python programming language, using well known libraries for data processing and analysis, such as numpy, pandas, scikit-learn, etc.

Deployment in Serbian pilot SCC/EMSS

As it can be observed in Figure 7, the deployment of the enhanced version of CARMEN in the Serbian pilot will be done in their production environment. Connection points are still to be determined. In the first phase, CARMEN will be connected to SCC's IT network, but in further stages of the project, CARMEN will be connected to the Control and Supervision network to analyse specific protocols and the rest of the industrial network traffic. The possibility to connect CARMEN to the Field network will also be considered and studied but, due to the previously mentioned problems that this connection may cause, this option will only be used if it is feasible.

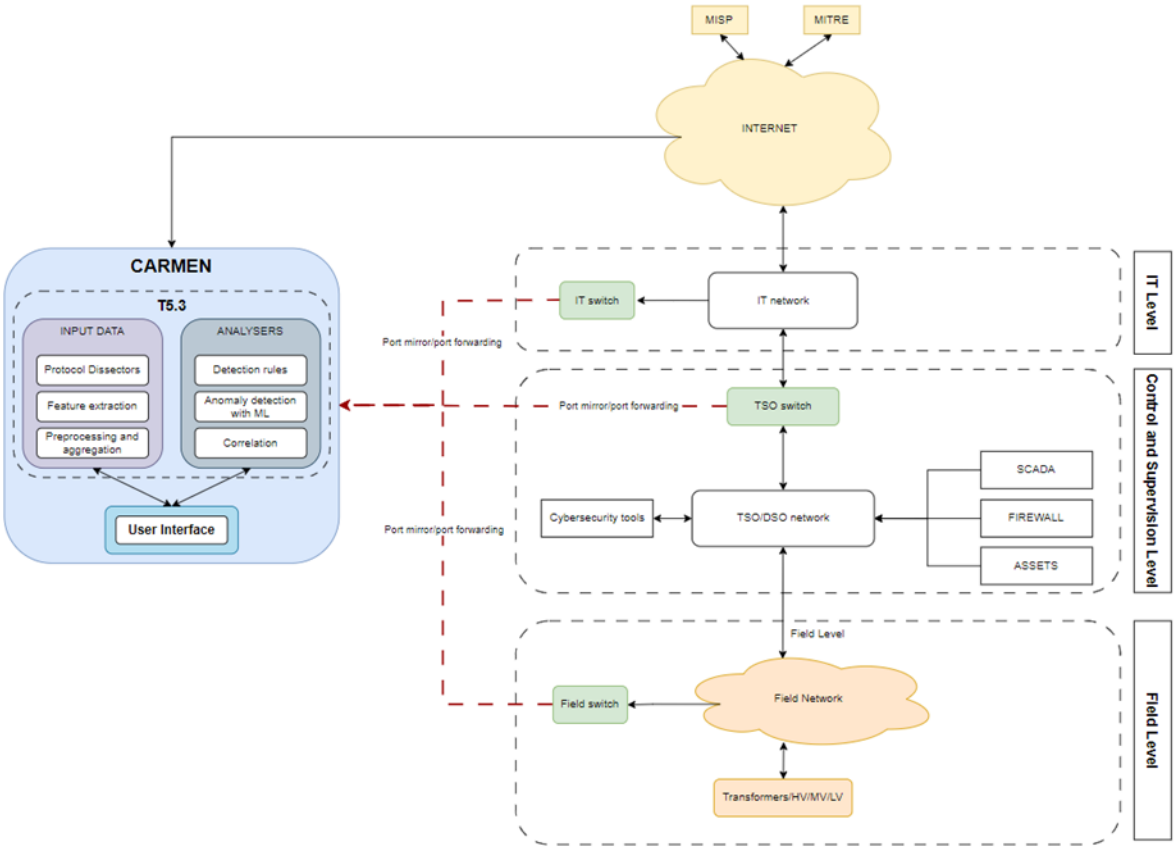

Figure 7 – UC33 component layer diagram for Serbian pilot

Deployment in Greek pilot HEDNO

As it can be observed in Figure 8, the deployment of the enhanced version of CARMEN in the Greek pilot will be done in their pre-production environment in UC27, called Sandbox tool. This deployment will be used to test both UC33 and UC34 use cases. CARMEN will be connected to one of the switches within the network to analyse all the possible traffic from the assets by means of using port mirroring and port forwarding. After that, if possible, the number of assets should be increased to detect more anomalies.

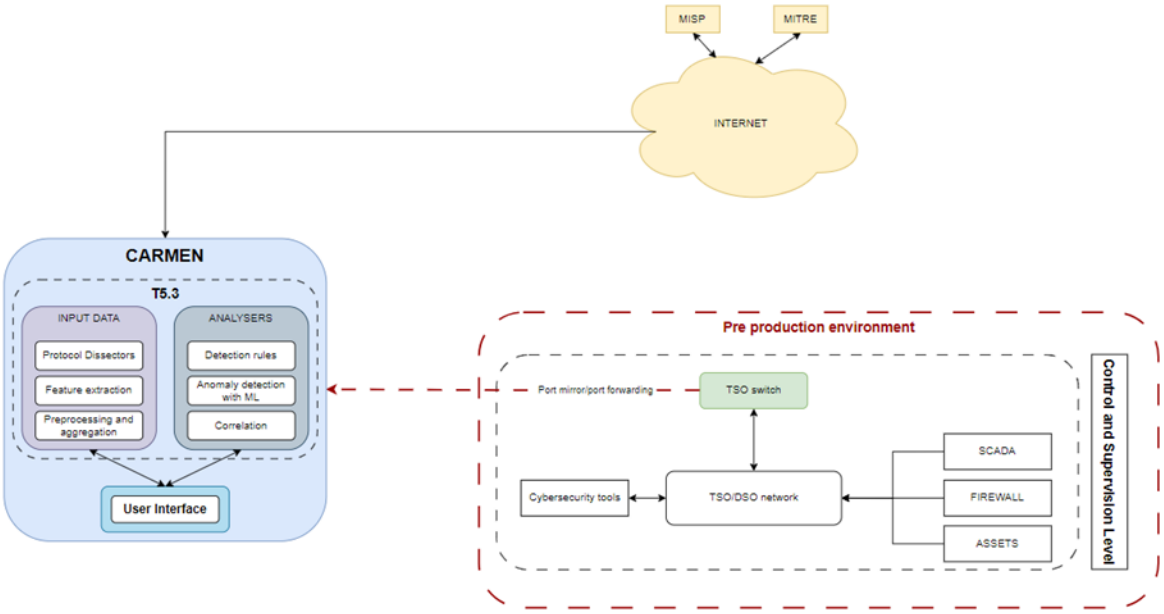

Figure 8 – UC33 component layer diagram for Greek pilot

Deployment in Slovenian pilot ELEK

As it can be observed in Figure 9, CARMEN won't be connected to any network from ELEK because of security policies, so the way we gather data will be using traffic captures from some networks. This deployment will be used to test both UC33 and UC34 use cases. The first phase will involve 1 day traffic capture to analyse data and get some previous view from the networks. After that, in a second phase, we will request more captures to start analysing for anomalous actions to get some possible alerts.

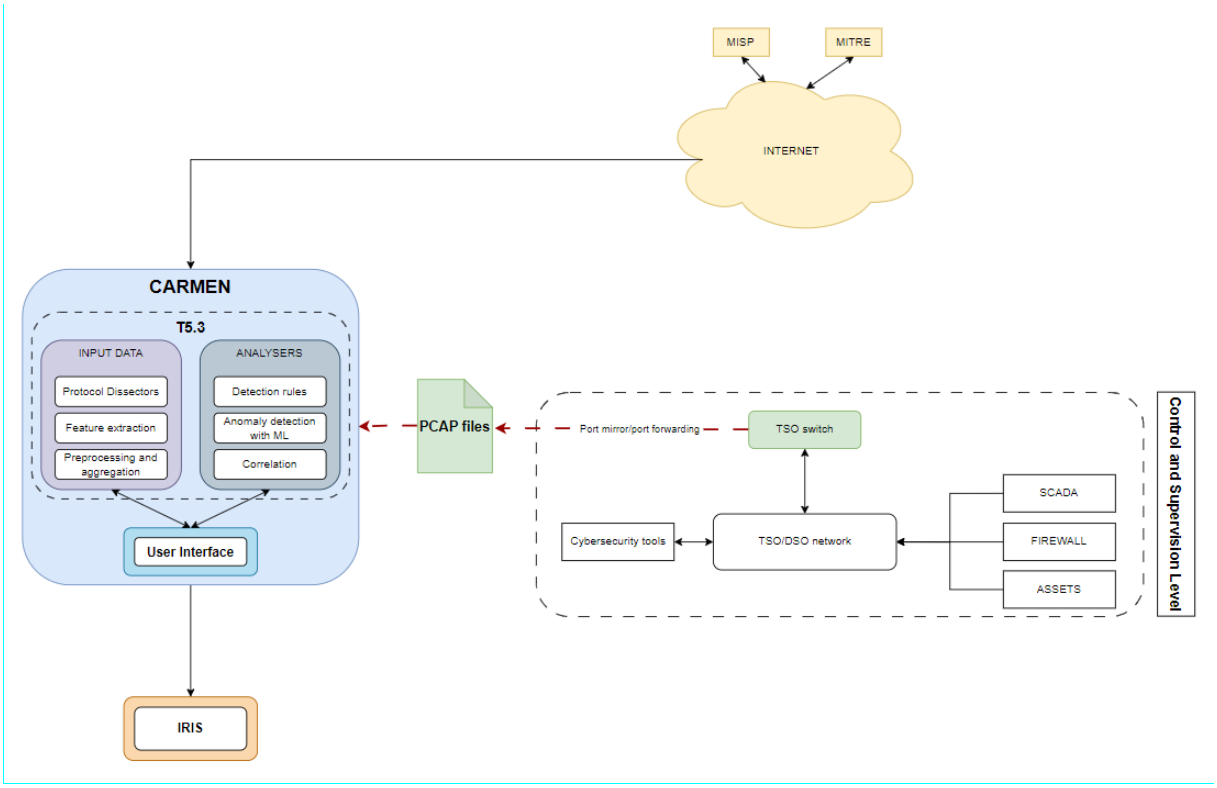

Figure 9 – UC33 component layer diagram for Slovenian pilot

4.3.2 UniFusion platform

The aim of the UniFusion platform is to achieve real-time telemetric data processing with secure data exchange with establishment of secure communication protocols using the latest protective communication measures such as data encryption, access filtering from allowed IPs. The UniFusion system will perform automatic data checks and detection for attempts of unauthorized data read, receiving data outside the expected limits, and communication requests from unapproved addresses. This tool will be tested in collaboration with the Slovenian pilot.

Internal architecture of the tool
Figure 10 shows the architecture of the UniFusion platform concept.

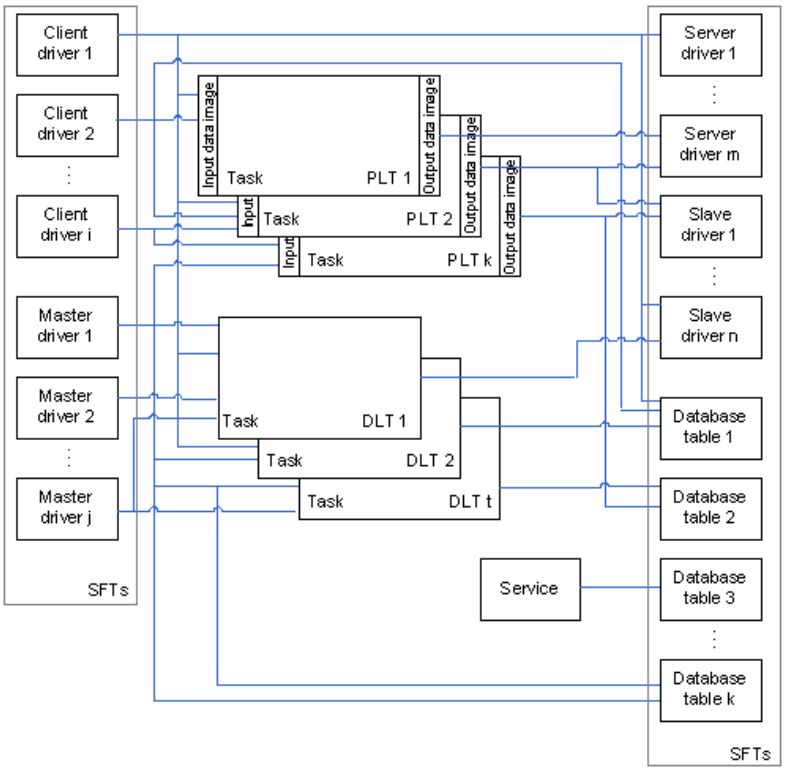

Figure 10 – UniFusion platform concept

Example of a communication gateway with the UniFusion platform is in Figure 11.

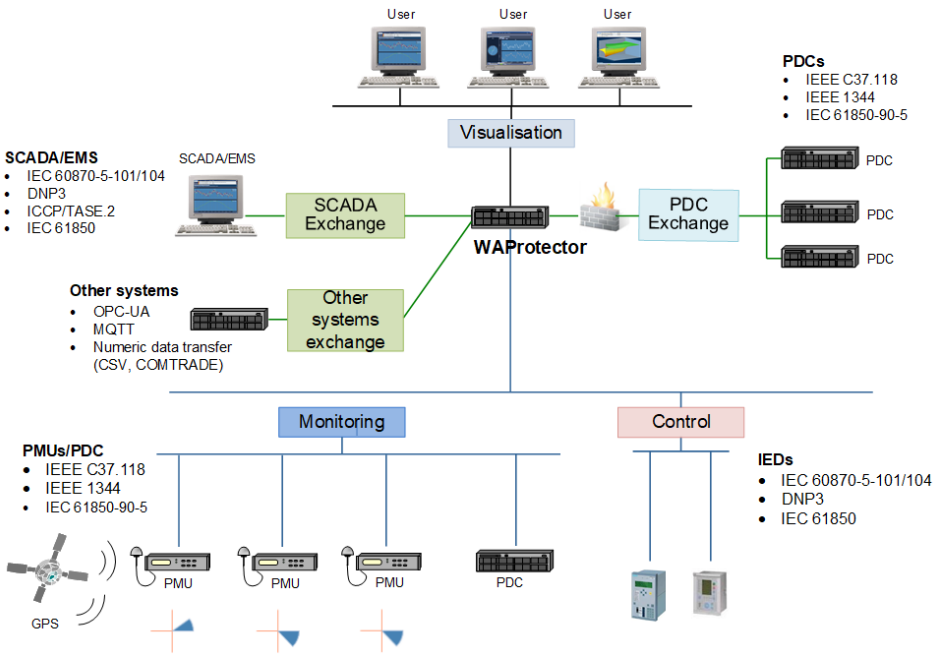

Figure 11 – Communication gateway with UniFusion platform

Figure 10 presents the architecture of the UniFusion platform with input communication drivers on the left side, programmable tasks in the middle and output communication drivers on the right side.

Database and human machine interface (HMI) support are parallel parts of the real-time processing. UniFusion platform is developed on .NET technology and can operate on Windows or Linux operating system. Software is developed for parallel processing. Communication drivers by standard communication protocols are implemented for data exchange inside the power systems.

Standard encryption algorithms will be used to increase the security. Data exchange will be provided from:

- SCADA
- Consumers
- DER

Protocols used for data exchanges are:

- ICCP/TASE.2
- MQTT
- IEC 60870-5-104

Data flow used in the system as an example in Figure 12 will be from:

- SCADA center (ELEK) : ICCP
- Measurement data centre (ELEK): MQTT
- Flexibility system (ELEK): ICCP, MQTP

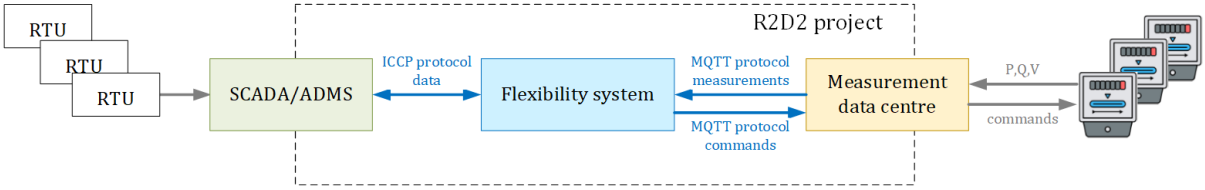

Figure 12 – Example of data flow as used in UC10

User interface

Visualisation will present monitoring of the communication statuses and data flows. Data can be presented in tabular formats, charts, organised as customised dashboards.
During the R^2D^2 project the HMI will be configured with data received from smart meters, SCADA data and results of algorithms based on that data. Details will be presented in the next deliverable (D5.2).

Resources

Software resources needed for the development of this tool are:

- Standard .NET Framework is a proprietary software framework developed by Microsoft that runs primarily on Microsoft Windows. It was the predominant implementation of the Common Language Infrastructure (CLI) until being superseded by the cross-platform .NET project. It includes a large class library called Framework Class Library (FCL) and provides language interoperability (each language can use code written in other languages) across several programming languages. Programs written for .NET Framework execute in a software environment (in contrast to a hardware environment) named the Common Language Runtime (CLR). The CLR is an application virtual

machine that provides services such as security, memory management, and exception handling. As such, computer code written using .NET Framework is called "managed code". FCL and CLR together constitute the .NET Framework.

- UniFusion platform is multifunctional engineering platform for complex real-time data processing.

4.4 TASK 5.4 – DEEP LEARNING DATA ANALYTICS FOR SECURITY

In T5.4 S2 Grupo is improving and developing new capabilities for CARMEN. As explained earlier in this document, these capabilities will be validated in use case UC34 (see Figure 13). Developments in the scope of task T5.3 will provide CARMEN with new capabilities for acquisition and normalization of tactical and operational threat intelligence. Also, within the scope of this task, new ML-based capabilities will be developed for clustering all this normalized threat intelligence and for comparing new observed threat intelligence with the clustered database. In this way, the potential presence of APTs in the system can be assessed based on how similar a suspicious behaviour/activity observed in the system is to already known APTs'. Beyond the mentioned unknown threats' detection, these developments will also help analysts finding other malicious activity which may have been overseen and assessing potential risks and cascading effects. This tool will be connected to SCC and EMSS based on Serbian infrastructure. The improved version of CARMEN will also be installed in the pre-production environment built by ICCS and CYBER for the Greek pilot. In addition to that, results of this task will contribute to use case UC10 in the Slovenian pilot, in collaboration with ELEK and ELPROS, although in use cases related to Slovenian pilot, ELEK and ELPROS will use their own technology are also collaborating in this use case.

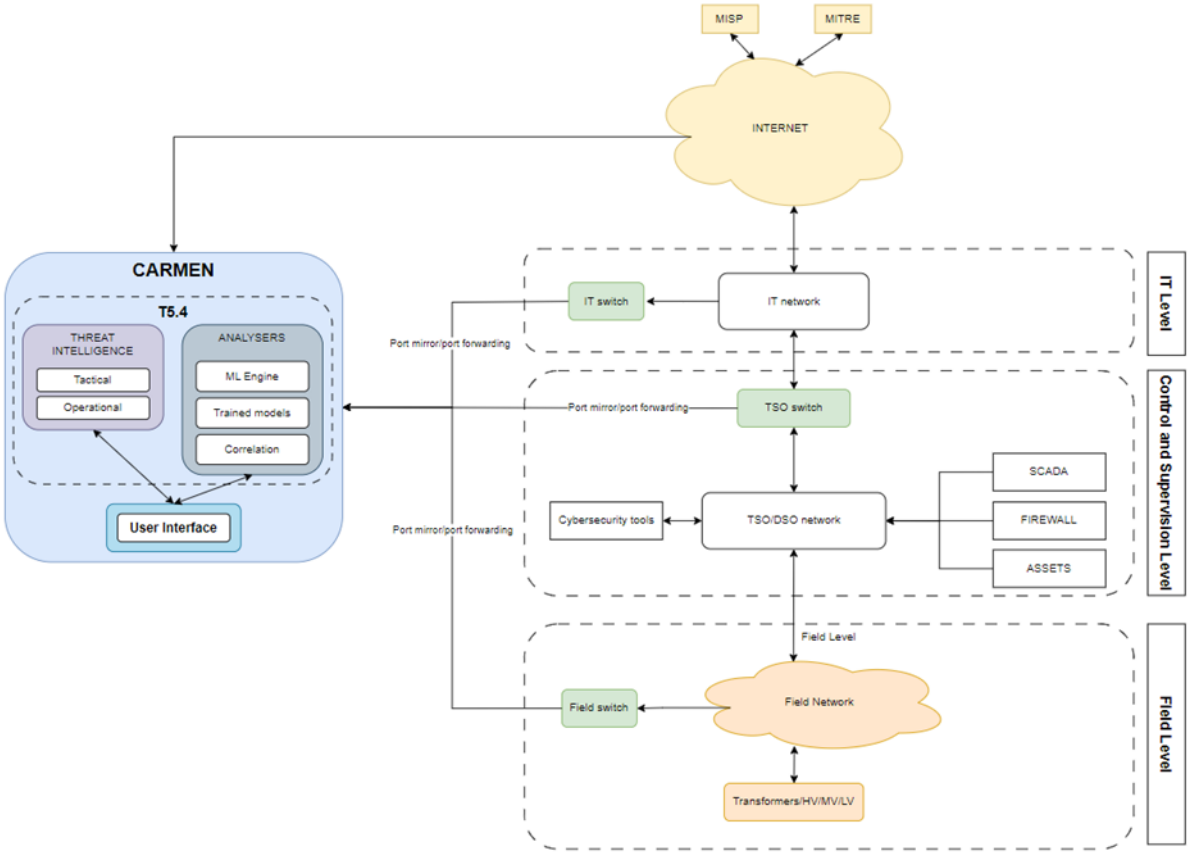

Figure 13 – UC34 component layer diagram

4.4.1 CARMEN tool in UC34

As in task T5.3 and use case UC33, the tool that S2 Grupo will use for covering use case UC34 is also CARMEN [7], the company's APT compromise detection tool, able to detect different anomalies and misuse. Within the scope of this task, S2 will enrich CARMEN's capabilities for threat intelligence ingestion and normalization from different sources (MISP [9], MITRE [8], previous cyberattacks, etc.). Once normalized, all this gathered information can be used for training different ML algorithms for pattern detection and correlation to detect new potential threats based on their similarity to these previously ingested already known threats.

Since this detection is carried out based on threat similarity, the first step characteristics, already known threats will be characterised and clustered combining ML algorithms, attending to the tactical and/or operational intelligence these threats use. After this initial clustering, when any suspicious behaviour is observed, it will be possible, not only to consider the presence of an APT in the system attending to its similarity to already known threats, but also to predict still unobserved/undetected behaviours or future cascading effects, based on this similarity. As a result, it is possible for cyber-security teams of final users to carry on an early detection of threats and to advance possible recovery actions in case these threats succeed.

As it can be observed in Figure 13, the deployment of CARMEN will be the same for these developments as for use case UC33. Once deployed in a real environment, the system's network traffic will be acquired by CARMEN using port mirroring, without generating any traffic

in the network of the final system and without interfering with its normal functioning or the quality of the service it provides.

The figure shows three possible points of connection (explained in more detail in Section 4.3):

- IT Network Level
- Control and Supervision Level
- Field Level

Main components developed or enhanced in T5.4

In the scope of task T5.4, CARMEN's capabilities will be increased to cover the project use cases with the development of:

- Tactical and operational threat intelligence acquisition and modules: On the one hand, tactical intelligence is incorporated, focusing on understanding the specific methods and techniques employed by threat actors to infiltrate systems or compromise data. On the other hand, operational intelligence complements this by providing broader context, including the infrastructure, command and control mechanisms, and resources utilized by these malicious entities. This multidimensional approach allows the creation of robust threat models, representing various threat groups. The next steps involve combining multiple algorithms.

- Normalization modules: For the different ML algorithms to operate on the discussed threat information, there will be different modules in charge of applying various natural language processing (NLP) techniques that convert the text of their descriptions into numeric variables. NLP techniques focus on transforming natural language into a formal, programming-like language that computers can process. Several NLP algorithms will be considered for this preprocessing, such as Word2vect, Bag-of-words or Tfidf (Term frequency – Inverse document frequency), until the one that gives the best results is found.

- Threat intelligence clustering modules: Once the numerical characterisation of the threats is done, the clustering microservice groups them by similarities and common patterns. It will use the Birch algorithm which, through its "threshold" parameter, allows to be more or less demanding in clustering. The initial calibration of the algorithm parameters for testing will be fine-tuned through several iterations with the security team, but these parameters can be reconfigured to better suit each working dataset. All this process will take place offline, storing the clustering models for the next microservice.

- Threat likelihood modules: It will start from the threat clustering provided by the Birch model, against which it compares the information collected from the different sources in real time. In fact, this microservice calculates the distance of the analysed observation to each known group and generates an alert hypothesis when it is too close to any of the malicious groups. The cosine distance is the metric of choice for measuring the similarity between points in the vector space. It is particularly useful when dealing with high-dimensional data and is commonly employed in various fields such as information retrieval, natural language processing, and recommendation systems. The similarity is determined by the cosine of the angle between the two analysed vectors; the closer the cosine value is to 1, the more similar the vectors are, and the closer it is to -1, the more dissimilar they are.

Data exchanges, communication with other tools and/or products

Data exchanges in these developments will be the same as in developments made in task T5.3, since both tasks' contribution will aim at enriching CARMEN and will be tested deploying CARMEN in final pilots. Thus, more detail of this can be found in Section 4.3.1.

User interface

Regarding the user interface, it will be the same as in task T5.3, which is detailed in Section 4.3.1.

Resources

Resources necessary for developing and deploying the enhanced version of CARMEN are detailed in Section 4.3.1.

Deployment in Serbian pilot SCC/EMSS

As already commented and shown in Figure 14, the deployment of the enhanced version of CARMEN in the Serbian pilot will be done in their production environment. This deployment will be used to test both UC33 and UC34 use cases. Connection points are still to be determined. In the first phase, CARMEN will be connected to SCC's IT network, but in further stages of the project, CARMEN will be connected to the Control and Supervision network to analyse specific protocols and the rest of the industrial network traffic. The possibility to connect CARMEN to the Field network will also be considered and studied but, due to the previously mentioned problems that this connection may cause, this option will only be used if it is feasible.

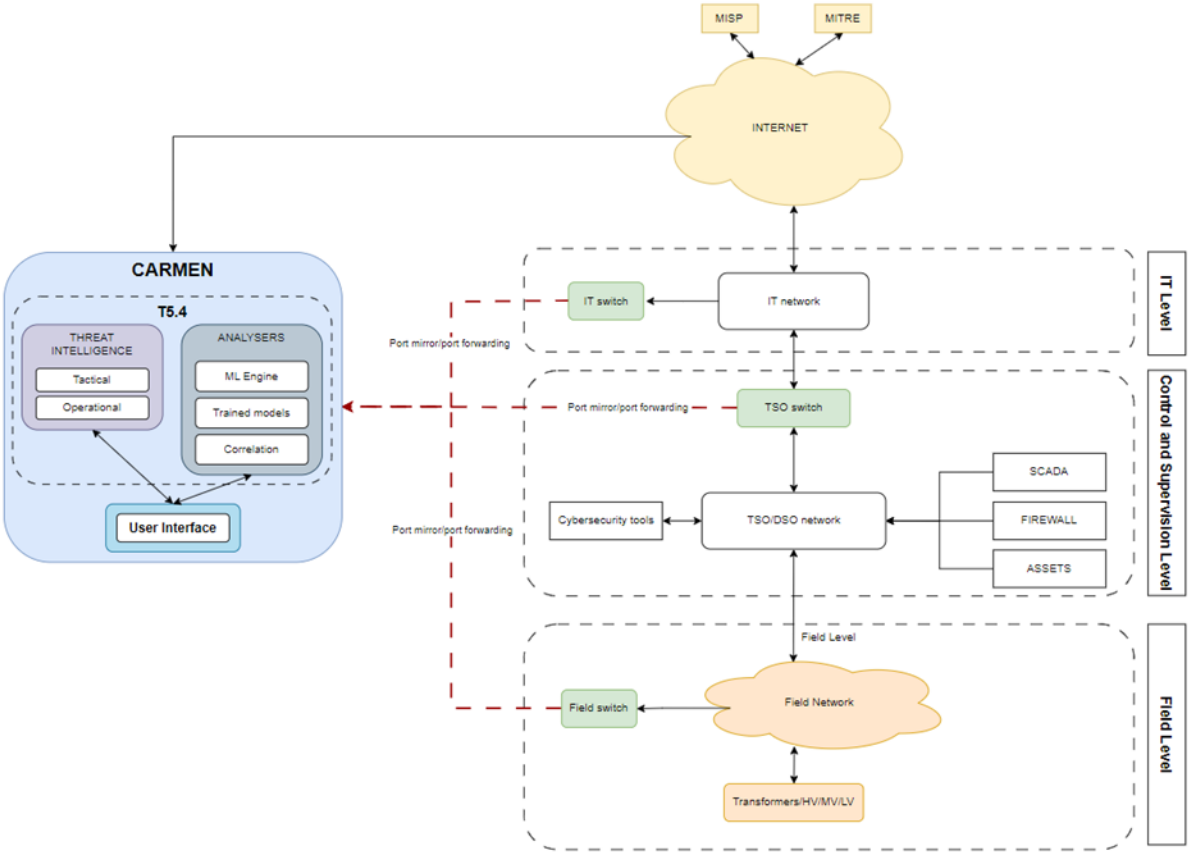

Figure 14 – UC34 component layer diagram for Serbian pilot

Deployment in Greek pilot HEDNO

As it can be observed in Figure 15, the deployment of the enhanced version of CARMEN in the Greek pilot will be done in their pre-production environment in UC27, called Sandbox tool. This deployment will be used to test both UC33 and UC34 use cases. CARMEN will be connected to one of the switches within the network to analyse all the possible traffic from the assets by means of using port mirroring and port forwarding. After that, if possible, the number of assets should be increased to detect more anomalies.

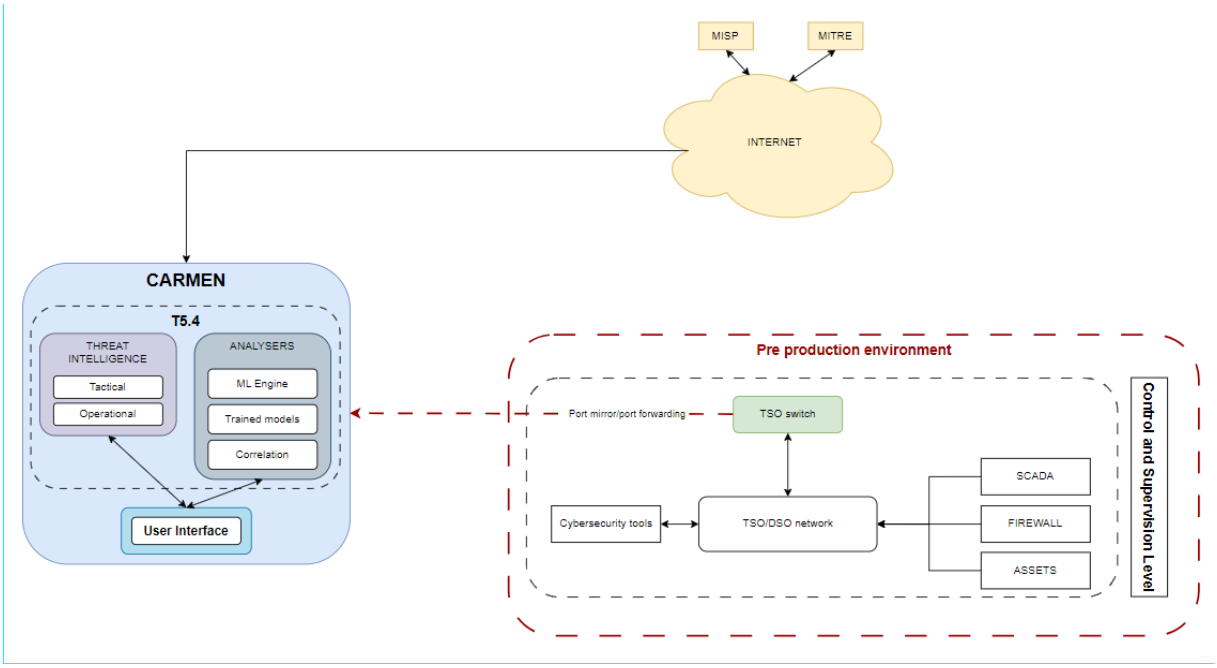

Figure 15 – UC34 component layer diagram for Greek pilot

Deployment in Slovenian pilot ELEK

As it can be observed in Figure 16, CARMEN won't be connected to any network from ELEK/ELPROS because of security policies, so the way we gather data will be using traffic captures from some networks. This deployment will be used to test both UC33 and UC34 use cases. The first phase will involve 1 day traffic capture to analyse data and get some previous view from the networks. After that, in a second phase, we will request more captures to start analysing for anomalous actions to get some possible alerts. This information will be also useful for the IRIS product as is related with ELPROS remaining use cases.

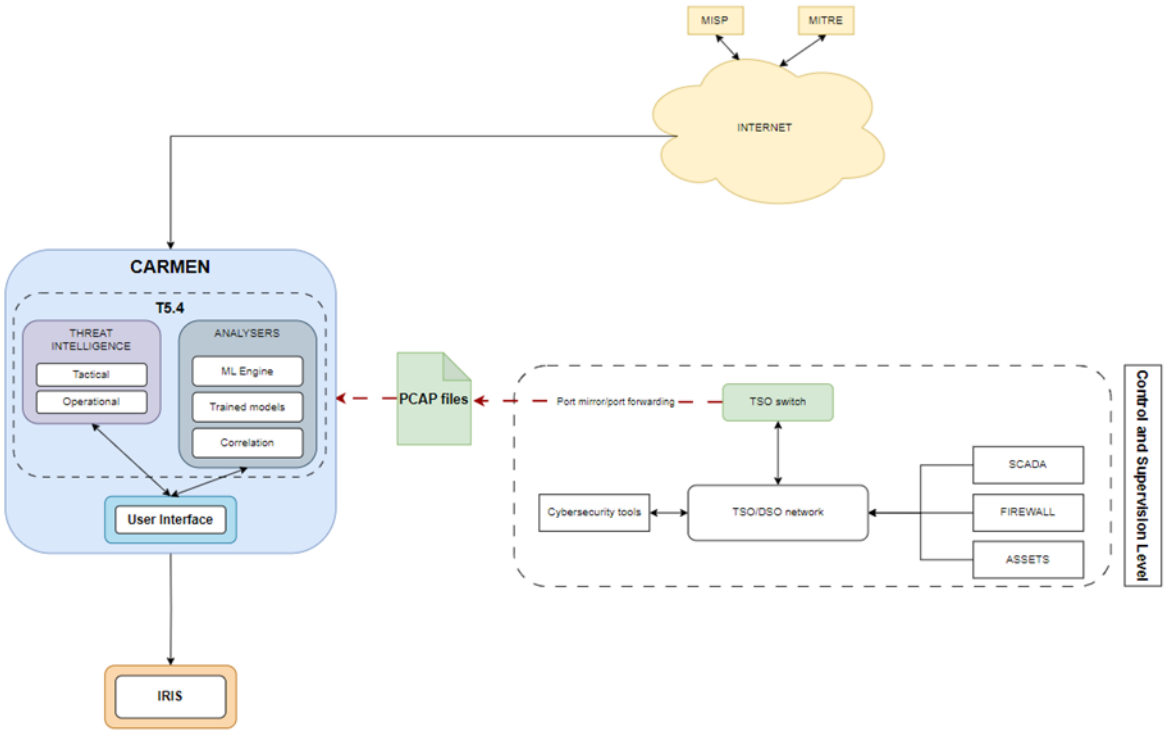

Figure 16 – UC34 component layer diagram for Slovenian pilot

4.5 TASK 5.5 – DEVICE ORIGIN AND SUPPLY CHAIN

The device origin and supply chain tools aim to provide spare part management solutions, end to end chain of custody of network hardware devices, versioning and updates on the grid firmware. More specifically, in the context of this task, two tools will be developed:

- **The Sandbox tool**: it will contribute towards the secure integration of new components by monitoring and classifying their behaviour before integrating them in the production environment,

- **The Self-assessment tool**: it will provide best practices and guidelines for the supply chain management process to contribute towards third party product integrity, by considering the best coding and development practices, quality assurance standards and processes, and anti-tampering requirements.

During this task, best practices and guidelines for the supply chain management process will be laid out to provide third party product integrity, by considering the best coding and development practices, quality assurance standards and processes, and anti-tampering requirements. The Sandbox tool will secure the integration of the new components by monitoring and classifying their behaviour before integrating them in the production environment.

Additionally, good vendor management practices that qualify and audit suppliers, will be defined to be adopted by operators.

The Self-assessment tool aims to help EPES operators improve the security of their supply chain practices and identify any possible gaps in their security controls regarding their supply chain practices. The assessment will also help EPES to identify any possible gaps in their security controls regarding their supply chains. Once these gaps have been identified, the EPES operator can develop a plan to implement the required controls. Additionally, good vendor management practices that qualify and audit suppliers, will be defined to be adopted by operators. This will assist EPES operators identify related security risks, reduce the risk of cyberattacks, improve resilience to disruptions, and protect their critical assets.

4.5.1 Sandbox tool

The Sandbox tool will contribute towards the secure integration of new EPES components and releases (with pilot partners HEDNO and ICCS), to detect and respond to abnormal and/or suspicious communications that might indicate malicious activity. The tool will be integrated with the EPES staging environment, such as an existing test-bed for newly acquired devices or releases for monitoring the behaviour of the newly deployed components. This module will utilize the functionality provided by T5.4 (Deep Learning data analytics) to identify attack vectors related to the supply chain by comparing normal communication patterns with the new component's behaviour. The use of blockchain technology (from T5.1) will also offer the necessary transparency and accurate end-to-end tracking in the supply chain process and will help to detect and prevent the introduction and use of counterfeit and fraudulent software releases.

Aim of the tool

The Sandbox tool can play a valuable role in detecting and responding to abnormal or suspicious communications that may indicate malicious activity. The following paragraphs provide a short description of the sandbox's contribution to this objective. With the deployment of the Sandbox tool, EPES operators introduce an additional layer of defence, enabling organizations to ensure the integrity and security of the components they deploy in their critical infrastructure.

Combining the Sandbox tool with blockchain technology, EPES organizations can establish a transparent and secure supply chain process that minimizes the risk of counterfeit and fraudulent software releases. The immutable nature of blockchain ensures the integrity of information, while the Sandbox tool provides the testing and monitoring capabilities necessary for effective risk detection and prevention.

Internal Architecture (Overview)

The Sandbox tool operates in a controlled testing environment where new EPES components and releases can be deployed and monitored, ensuring that any potential malicious activity is identified and contained within the sandbox. The architecture and the operation steps of the tool are depicted in Figure 17.

The initial step is the behaviour analysis of the existing components within the Sandbox tool, to establish baselines for normal communication patterns and identify any deviations or suspicious activities.

Traffic monitoring and analysis of network traffic generated by the new EPES components is analysed for anomalies or indicators of malicious activity, such as unusual data transfers, unauthorized access attempts, or unexpected communication patterns. Malware detection, by

analysing new components interactions and monitoring for any signs of malware presence or activity, is also feasible. In case the Sandbox tool detects abnormal or suspicious communications within the staging environment alerts, further response will be made available.

Over time, the Sandbox tool can learn from its observations and interactions with the EPES components to refine its detection algorithms and become more effective in identifying abnormal or suspicious communications.

The Sandbox tool will receive alerts and other information from the CARMEN tool, provided by S2 Grupo and enhanced with data acquisition and aggregation, as well as with threat detection and threat modelling capabilities within the scope of tasks T5.3 and T5.4 of the project. CARMEN tool is a SIEM (Security Information and Event Management) tool. Its capabilities, as well as the already mentioned developments and enhancements carried out in the project, are better described in sections 4.3 and 4.4 of this document.

The Sandbox tool will utilise the services provided by the R^2D^2 blockchain (T5.1), to provide transparency, accurate tracking, and prevention of counterfeit and fraudulent software releases, in supply chain management. The combination of the Sandbox tool and blockchain can address the following challenges:

- Transparent and Immutable tracking of the software releases
- Distributed Consensus and Trust
- Immutable Audit Trail of software releases and related assessment activities
- Enhanced Security and Anti-Counterfeiting
- Collaboration and Information among supply chain participants

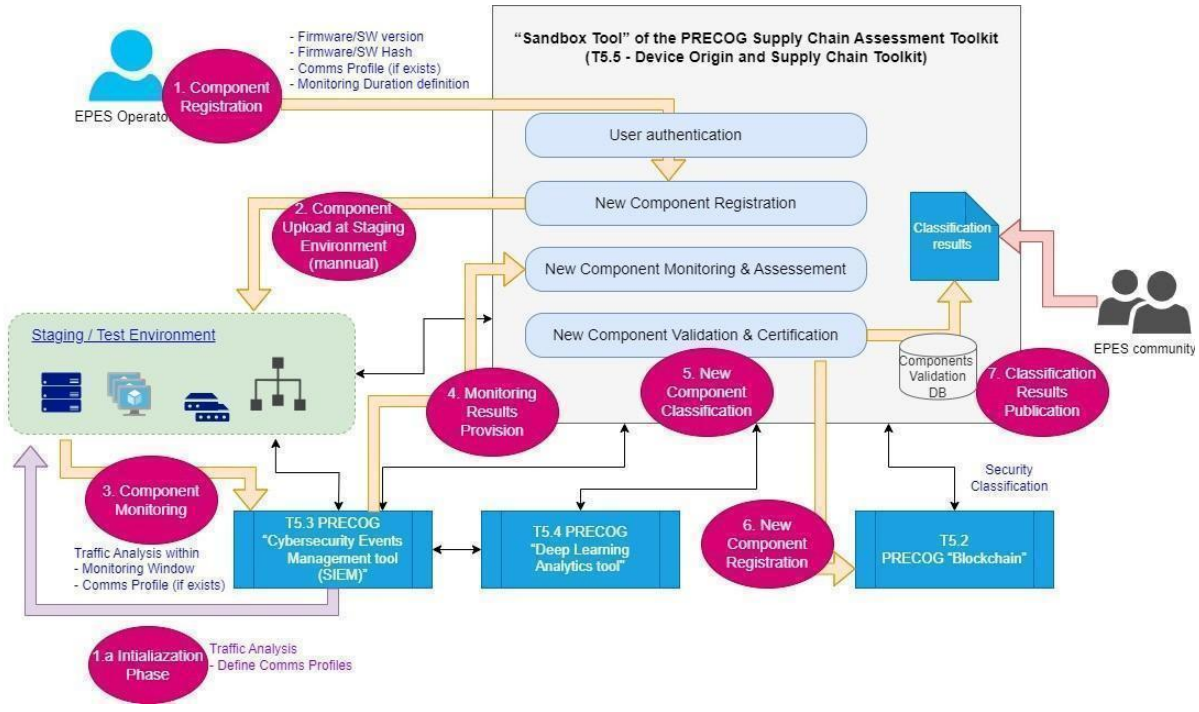

Figure 17 – Sandbox tool process flow

The Sandbox tool's basic process flow (shown in Figure 17) includes the following steps:

1. Using the web-based interface, the EPES operator registers a new component together with its parameters (version, hash, comms profile etc.) assuring CPE compatibility and the communication profile – if it is available.

1.a. An Initialization Phase using the tools provided by T5.3 and T5.4 records traffic generated from an existing component/device to define the traffic baseline. The traffic will be generated over a specific test scenario. This baseline will be used to identify traffic deviations for the new component being assessed, denoting malware existence.
2. The EPES Operator manually uploads/integrates the new components at the Staging Environment.
3. For a specific Monitoring-Time-Window (e.g. one week) the cyber-security Events Management tool (T5.3) will record the traffic generated by the new component/device to identify traffic deviations from traffic baseline, denoting malware existence.
4. The Deep Learning Data Analytics tool (T5.4) will return the outcome of the process to the Sandbox tool denoting if the component should be considered suspicious or trusted, as well as any other helpful data to be used to further classify the software.
5. The Sandbox tool uses the provided information to classify the software trustiness /credibility.
6. The analysed information will be used to sign the new software component's hash digest using the R^2D^2 Blockchain's functionality (T5.1).
7. The Sandbox tool will store the results and any other useful data (name, version, date, signature, etc) in a database accessible to the EPES community to compare their software with the tested ones.

An example of the Sandbox tool operation is described in ANNEX I

Figure 18 depicts the proposed architecture of the Sandbox tool of the PRECOG Supply Chain Assessment Toolkit.

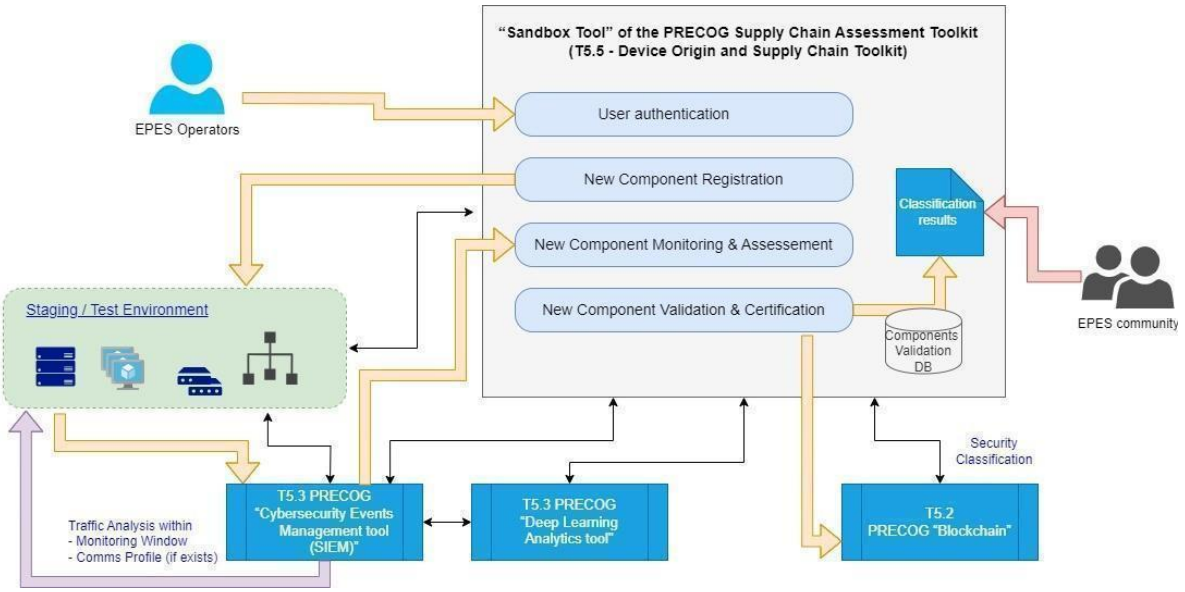

Figure 18 – Sandbox tool architecture components

The Sandbox tool is a web-based software tool architected to provide the following capabilities:

User authentication

Authentication is one of the fundamental pillars in cyber-security, since it verifies a user or device before allowing access to a system or resource. Access to the Sandbox tool software capabilities will be authenticated.

New Component registration

Authenticated users have the ability to register a new component to be tested. User will input data and the software hash if it is available from the vendor. The data that describe the new component are aligned with the Common Platform Enumeration (CPE) standard. CPE is an industry standard used to provide a uniform way to show information for systems, software and packages. Based upon the generic syntax for Uniform Resource Identifiers (URI), CPE includes a formal name format, a method for checking names against a system, and a description format for binding text and tests to a name. It can also be used for software and hardware inventories, as well as for enhancing vulnerability management to track results among different products.

Using the Sandbox tool, the user will calculate the hash digest using the SHA_256 hash algorithm.

Baselining

To compare the new component's traffic, CYBER needs to have a legitimate traffic sample – the baseline. The baselining has two options:

1. The first one concerns an existing component that has been updated with a firmware/patch/update. In this case, users can upload/import traffic data generated from an existing component to define the traffic baseline, the legitimate communication profile – generated from current usage scenarios. The traffic is collected using the data acquisition and aggregation capabilities developed within the scope of T5.3 for the CARMEN tool, described in section 4.3.
2. The second one covers the introduction of a new component. In this case, the users must describe the communication profile, meaning the legitimate communication paths –, define the test scenarios to be executed, integrate the new component into the staging environment to collect the generated traffic. The generated traffic for a defined period of time is collected and analyzed with the CARMEN tool, using both their previously developed analysis/detection capabilities and those enhancements carried out within the scope of tasks T5.3 and T5.4.

Traffic Analysis

Users will install the new EPES component to the staging environment and start testing the new component using an approved usage (test) scenario for a defined period of time. During this period, the generated traffic is monitored, collected and analysed using the version of the CARMEN tool improved in tasks T5.3 and T5.4. Results and traffic data generated from the new component are imported into the Sandbox tool database (Components Validation DB).

Classification

As “Secure” CYBER classifies a component that during the defined test period, in the isolated staging environment and for the approved usage scenario, if there is neither suspicious traffic, collected and analysed by the Sandbox tool, nor the collected traffic deviates from the baseline.

As “Suspicious” CYBER classifies a component that during the defined test period, in the isolated staging environment, for the approved usage scenario, there is suspicious traffic, collected and analysed by the Sandbox tool and/or deviations from the baseline are detected. In such a case, the new component is sent back to the vendor for further monitoring, root cause analysis and remediation, if required. The vendor may answer the results of the “Suspicious” classification, justifying the collected traffic or releasing a software update (patch).

Signing

Regardless of the classification result, “Secure” or “Suspicious”, hash digests of the new component’s software or firmware binaries along with hash digests of the classification data will be registered using the blockchain tooling capabilities of the T5.1 tool where hash digest are registered on blockchain instead of storing data itself. Using blockchain technology (described in subsection 4.1.1), the Sandbox tool provides a transparent, decentralised, distributed trust, immutable audit trail and enhanced security and anti-counterfeiting measures.

Results Dissemination

A new component acquired from an EPES is highly likely to be acquired from many more EPES operators. In order to reduce the risk of using an infected item, preserve money and time, the classification results and the signed hash are stored in the Components Validation database for further usage and reference from the EPES community. Using this repository of tested components results, any verified and registered user can query the database to find if the component has already been tested and classified.

The Sandbox tool consists of the following components:

Staging Environment

The staging environment (also referred as “staging”) is an isolated testing environment consisting of EPES components that exist in the production environment. Usually, the staging environment can be a “miniature” of the production environment, or a digital twin, comprised of the minimum technology items to execute specific tests. The size and the content of the staging environment may change according to the specific needs of the case.
For cyber-security evaluation purposes, there is a need for a platform/test-bed where a wide range of cyberattack scenarios can be simulated. The staging environment, which is a testing infrastructure that emulates the functionality of a Supervisory Control and Data Acquisition (SCADA) system, serves exactly this purpose. In the next sections, the functionality and the communication architecture of the staging platform are analysed in detail. Before this, some preliminary terms are briefly explained to familiarise the reader with the platform. Finally, the threat scenarios that can be tested and evaluated in this environment are presented.
The components constitute the staging environment for the R^2D^2 project are as follows:

Preliminaries

- The functionality of a SCADA system can be summarized as follows: A SCADA system ensures the data transmission from monitored devices to controlled equipment (e.g. sensors, motors, etc.) and vice versa, while also illustrating the acquired information to an interface where the data can be analysed and utilized for reporting purposes. Typically, a SCADA system comprises Programmable Logic Controllers (PLCs), Remote Terminal Units (RTUs) and Human Machine Interfaces (HMIs). PLCs and RTUs are microcomputers that serve as local data aggregation sites, responsible for sending and translating data to remote or on-site HMIs, while also executing control commands to the field devices. The data are visible to human operators via the HMI, which can either be cloud-based or reliant on dedicated servers. The above description of a SCADA system yields that the staging environment emulates a highly crucial part of Smart Grid systems.
- MQTT is a standardized messaging protocol that is specially crafted to facilitate machine-to-machine communication. It is a standard option for Smart Grid infrastructures, where various modern microtechnologies, such as smart meters, intelligent electronic devices, PLCs, etc., operate within resource-constrained environments. MQTT enables bidirectional messaging among the interconnected devices and the cloud, of low-latency and limited bandwidth, which in turn, allows the direct and fast communication of the different Smart Grid components. MQTT operates under the subscriber/publisher communication architecture, where different publishers (smart meters) send their data to certain protocol brokers (dedicated software) and the subscribers (devices, software applications) of the protocol brokers consume the acquired data.

Operation & Architecture

An AMI infrastructure is deployed in the Greek Pilot site where several smart meters, called SLAMs, have been installed in various sites (Mesogeia, Kythnos, Athens city center, etc.). These SLAMs collect power system-related information from individual household loads, such as active power, voltage measurements, etc. Since this type of information involves personal data, their privacy is ensured with strong cyber-security measures and the user cannot view their information, despite their access to the infrastructure. The SLAMs are interlinked with each other and they can communicate since they are connected in the same VPN network. Furthermore, these SLAMs send data about the actual measurements that they collect and logging information about their status to a remote MQTT broker which is located in the same VPN network with them.

As the MQTT broker collects data from the SLAMs, a Python software application, which is located in the aforementioned VPN network, consumes the data that are published by the MQTT broker. Then, this Python application stores the data to different dedicated servers (SQL Server, MongoDB, etc.), located across the same VPN network. At this point, it should be noted that there are different clusters of SLAMs which correspond to different data collections of particular locations. For each SLAM cluster, a different database server is deployed to avoid the mixing of non-relevant data. After the information storage, these database servers expose their data to dedicated, web-based APIs which operate as the HMIs of the infrastructure. The previously described operation and architecture of the staging environment are clearly illustrated in Figure 19.

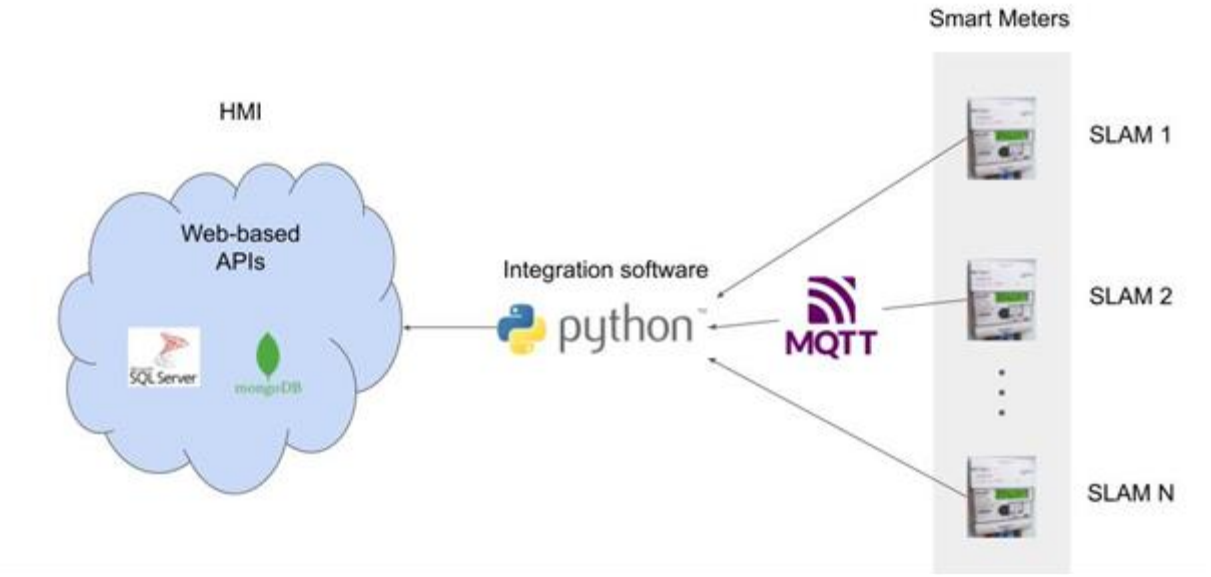

Figure 19 – Staging environment high level architecture

The functionality of the staging environment that has been described so far, corresponds to the part of the SCADA operation that collects data from field devices and plots them to an HMI for visualization and reporting. The decision-making part of SCADA which sends control commands to the proper devices for regulation of the physical systems has not been developed yet, but it will be completed in a more advanced stage. The control part of the staging environment will involve the communication of the deployed with various SEL controllers to regulate different types of loads (e.g. air conditions).

User authentication

Authentication is one of the fundamental pillars in cyber-security. The users to access the tool, can be created and maintained on the tool itself, and/or they can exist on an EPES directory or IDP. The integration of the tool with the external directories or IDPs is based on well-known protocols.

New Component Registration

The module that supports the registration of a new component is web based, accessible only from authenticated and authorized users. It has the following text fields to be filled in by the appointed user:

- Part
- Vendor,
- Product,
- Version,
- Edition,
- Update,
- Language,
- SW_edition,
- Target_SW,

- Target_HW,
- Other
- Type
- Serial Number
- Date
- Category
- Hash

It provides the capability to enter a new component. If the operator wants to register a new software release of an existing software can use the data already entered to create a new one. For example, if the release SW v1.4.5 has just arrived, the operator finds the data for the registered SW v1.4.4 that already exists, updates the fields for the new release and saves the new record.

For example, consider the scenario that a user wants to register a new software update for an existing component to be tested. The data are stored in the Components Validation DB. The fields: vendor, product, version, edition and update, comprise the unique identifier of a component.

If the hash digest of the software is not available, the user can calculate it using the Sandbox tool and then save for further usage.

Components Validation DB

Components Validation DB is the Sandbox tool repository. It is used to store:

- Components data
- Traffic baselines
- New components traffic data
- Classifications' results
- Configuration settings
- Application users

Monitoring & Assessment

To evaluate a new component, an Initialization Phase is required. During this phase a traffic baseline will be established, using either a communication profile provided by the vendor or using the S2 tools from T5.3 to record the traffic generated from an existing component/device at the staging environment. The traffic baseline should be established on the anticipated operational communication patterns. This baseline will be used by S2 (T5.3/4) to identify traffic deviations denoting malware existence.

EPES Operator/ICCS will manually upload/integrate the new component at the Staging Environment. For a specific Monitoring-Time-Window (e.g. one week) S2 (T5.3) will record the traffic generated from the new component/device to identify traffic deviations (from traffic baseline) denoting malware existence. Using the S2 T5.4 the traffic denoting if the component should be considered suspicious or trusted.

Validation & Classification

Validation is a process that involves testing a component to ensure that it meets a pre-determined specification. This requires that during the testing of that component it will perform as it is defined by the vendor, and it will produce the expected traffic.

The traffic analysis results, as well as any other helpful data, will be used by CYBER to further classify the software as “Secure” or “Suspicious”. A security analyst will be presented with the monitoring results, the machine learning results and the proposed classification for a classification decision.

Having the classification decision of the software, the Sandbox tool consumes a web service provided by GUARD (T5.2) over the internet to sign the hash digest along with the classification data.

The signed software hash digest, the proofs and classification results are stored in the Components Validation DB.

Classification Results Publication

To reduce the risk of using an infected component and reducing the effort of re-testing an already tested component, the classification results and the signed hash already stored in Components Validation DB are available to the EPES community.

A user that wants to access the results:

- Register herself to the Sandbox tool module that provides access to the results. The registration requires a valid email and a password.
- The module sends to the user a verification email and upon user verification, queries the database to find if the component has already been tested and classified.
- The registration process is required to keep track of the users accessing the results as well as avoid overwhelming the tool with anonymous malicious queries.

Techniques & Algorithms

The Sandbox tool will be built upon the latest technology standards in order to provide a modern and secure technological environment.

The tool will use:

- TLS 1.2 protocol to secure data in transit
- The SHA_256 hashing algorithm to compute the hash digest
- A hashing algorithm (like BCrypt, Argon) to anonymize the user passwords, if no external directory is used
- If an external directory or IDP is used, SAML over HTTPS or NTLM, or LDAPS is used.
- Dissectors for extract traffic information from industrial protocols (MQTT, ICCP 60870-6/TASE.2, IEC 60870-5-104, Modbus)
- Intelligence units (IOCs, analyzers to process the output of dissectors, python code to execute to detect anomalies at the endpoints)
- Cyber intelligence module (normalization, intelligence acquisition, APT clustering)

Data Exchanges & Interfaces

Sandbox tool will exchange data with the following R^2D^2 tools:

- The CARMEN tool (T5.3 and T5.4)
- The KSI tool (T5.1)

Regarding the S2 tools (CARMEN, Claudia, List, Argos), the communication shall be performed as follows:

- Collector: Retrieval of log records from the ML module to CARMEN (HTTPS/443).
- Tactical Intelligence: Retrieval of Atomic and Behavioral Indicators from MISP (HTTPS/443).
- Operational Intelligence: Get entries from MISP (HTTPS/443)
- All capabilities will be implemented with Python scripts.
- As for the ML module, once there is a collection of possible threats, knowing their behaviours and features, it is time to group them by similarities and common patterns. The process of characterizing and grouping known threats is done offline, meaning it occurs outside real-time monitoring. This way, it does not impact the performance of the threat detection system during its operation.

The Sandbox tool will consume a webservice over the internet to sign the software classification results.

- General: Web service hosted in GUARD's environment communicates with the blockchain so the only recruitment for Sandbox tool is to have access to CYBER`s web service (for registration and validation). The web service in the GUARD environment accepts JSON based requests, documentation for describing the required endpoints and its requests and responses will be provided. This service will provide any means to store the proofs.
- Registration: Data registration is done using hash (data is hashed at client side and only the hash digest is sent to GUARD to avoid any unnecessary data transfers). Upon successful request a proof is returned. That shall be stored in the client environment keeping the link between data that was registered and its proof. Upon unsuccessful request an error message with code is returned indicating what went wrong.
- Verification: During the validation process the data's hash digest shall be provided to the verification endpoint along with corresponding proof to validate the data. Upon successful verification a "verification OK" message will be returned. Upon unsuccessful verification an error message with code is returned indicating what went wrong.

User interface

The Sandbox tool will be a web-based environment. The following mock-up (shown in Figure 20) depicts the Sandbox tool log-in page.

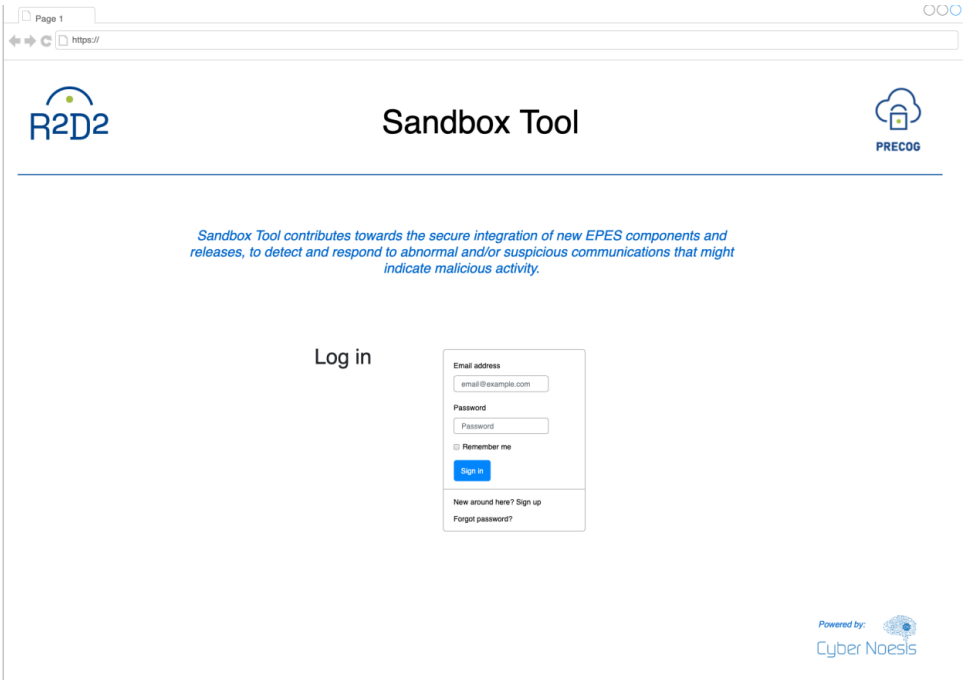

Figure 20 – Sandbox tool User authentication

The following mock-up (shown in Figure 21) depicts the Sandbox tool New Component registration page:

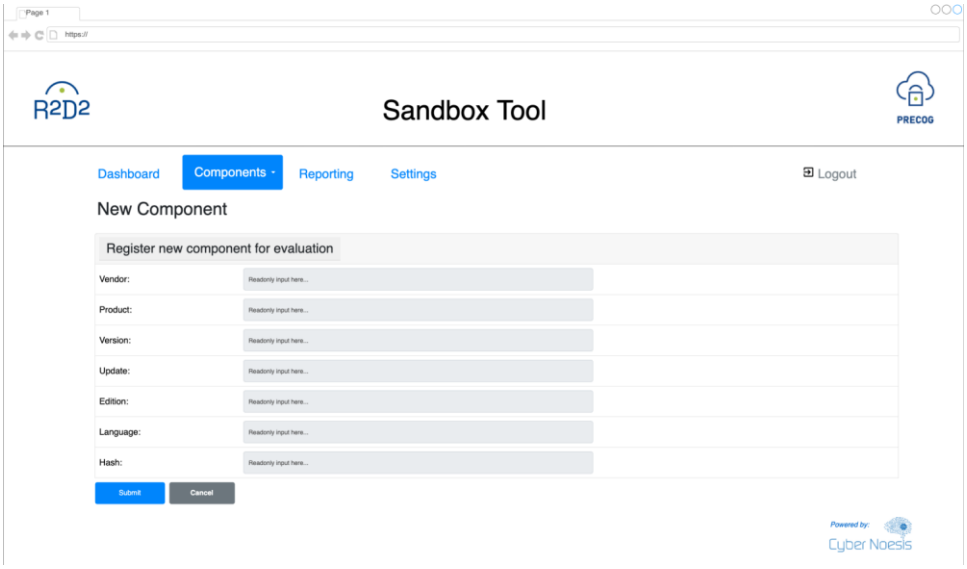

Figure 21 – Sandbox tool new component registration

The following mock-up (shown in Figure 22) depicts the Sandbox tool New Component classification page.

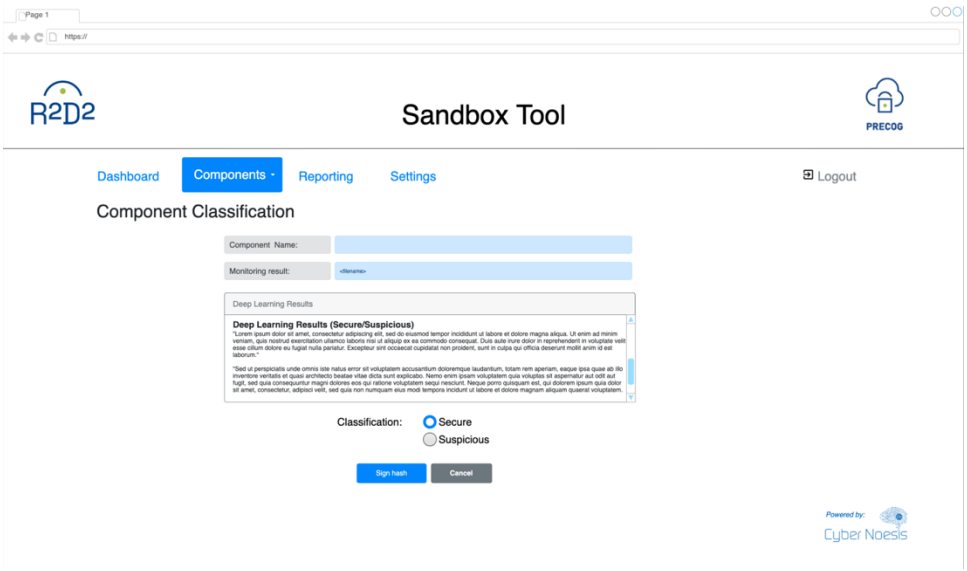

Figure 22 – Sandbox tool Component Validation & Classification

The following mock-up page (shown in Figure 23) depicts the Sandbox tool List of Components have been tested and classified.

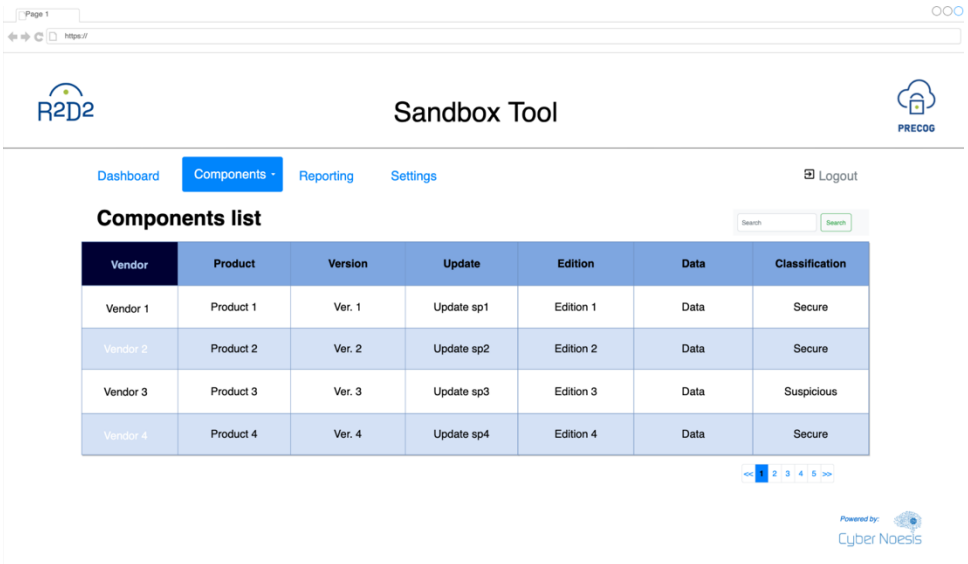

Figure 23 – Sandbox tool Classified Components' List

Resources

The Sandbox tool will use the following technology stack:

- Servers

    - A Virtual Machine will host the application server and the web server of the Sandbox tool
    - A Virtual Machine will host the DB to store
        - The collected traffic
        - The analysis results
        - The new components' information
        - The classification results
- Operating System
    - The operating systems will be Linux flavour
- Web server
    - The web server is an Apache HTTP Server 2.4.57
- Database management system
    - The Database management system is MySQL Community Server 8.0.34
- Development language
    - Python and HTML5 will be used
- Software Libraries
    - Python Django and Bootstrap css

4.5.2 Self-assessment tool

The Self-assessment tool can help companies identify security risks by conducting a security self-assessment of their supply chain process. This self-assessment can help EPES operators and vendors identify any possible gaps in their security controls regarding their supply chains. The tool can be used to assess the security of both physical and digital assets in the supply chain and can also identify potential risks at all stages of the supply chain, from the sourcing of raw materials to the delivery of finished goods to customers.
Moreover, the tool will be available to the vendors of the EPES operators to assess their security posture against standards and compare their performance against the average results of their community.
Additionally, the tool will be a central point of reference hosting the Supply Chain standards and guidelines and can be used to assess the effectiveness of existing security controls and identify areas where additional controls are needed, generate reports that can be used to communicate the results of the assessment to stakeholders, as well as track the progress of the company's security efforts over time.

The objectives of the Self-assessment tool are:

- Risks identification: The self-assessment tool helps EPES to identify and assess potential risks originating from their supply chains. By asking relevant questions about security measures, access controls, transportation, data protection, and other critical aspects of the supply chain, the tool highlights areas of potential concern.
- Regulatory adherence: The tool assists organizations in evaluating their compliance with supply chain security regulations, industry standards, and best practices.
- Gap analysis: Based on the answers provided, the tool identifies gaps or areas that require improvement in supply chain security. Reports or recommendations

help organizations prioritize and plan remediation actions to enhance supply chain security measures.

- Suppliers' evaluation: The self-assessment tool provides criteria to evaluate and assess the security capabilities of potential and existing suppliers. This can help organizations make informed decisions regarding supplier selection, ongoing monitoring, and risk mitigation strategies.
- Monitoring and improvements: Supply chain security is an ongoing process that requires continuous monitoring. The self-assessment tool can serve as a periodic check to measure the effectiveness of security controls and track progress over time.
- Awareness: The tool can also contribute to raising awareness among stakeholders and employees regarding supply chain security.

Internal architecture

The aim of the tool is to let EPES and providers to provide guideline about their security and a security posture measurement model. Having said that, the tool provides an environment to support the evaluation of the EPES supply chain management practices as well as their vendors. Moreover, the tool will help EPES to identify potential vulnerabilities and risks in their supply chain's cyber-security practices, in different technologies, such as IoT devices or cloud solutions.

The assessment uses a rating scale or a scoring system to quantify the cyber-security maturity in each area. The results will help identify areas that need further improvement and guide the development of a cyber-security strategy for the EPES supply chain. Regularly revisiting and updating the self-assessment tool is essential to stay ahead of evolving cyber-security threats.

The questions are grouped into domains. The EPES (Pilot) has the option to set up her own custom weights on the question domains to quantify the cyber-security maturity in each area.

More than one questionnaire may be released at the same time to be answered from EPES and Vendors. For example, one questionnaire may refer to supply chain management while the other one may refer to software development standards. The architecture and the operation steps of the tool is depicted on Figure 24.

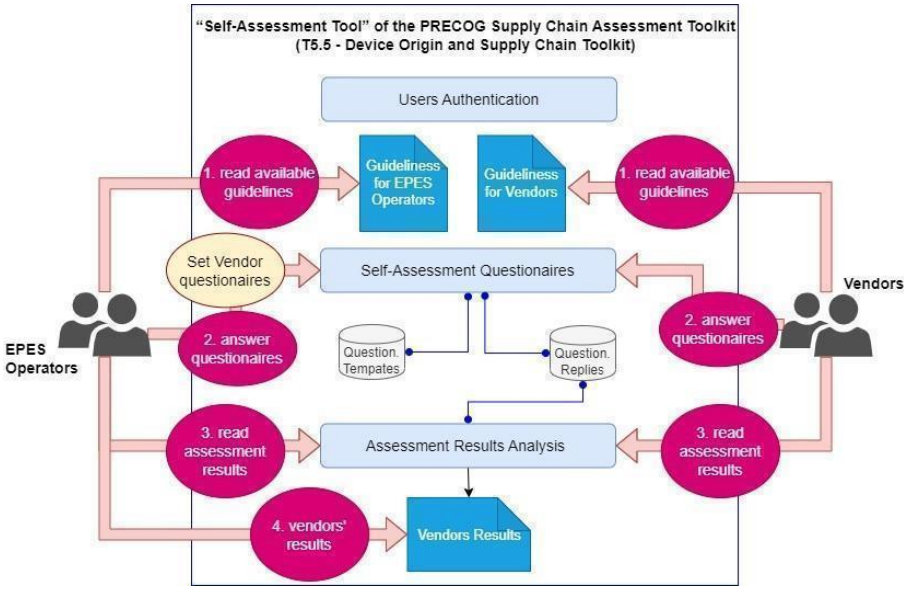

Figure 24 – Self-assessment tool process flow

More specifically the process flow consists of the following steps:

1. EPES operators & vendors read the appropriate guidelines, per their role
2. EPES operators & vendors answer their corresponding self-assessment questionnaires
    - Supplier chooses what questionnaire to answer
    - The questions are preset
    - Supplier starts answering the questionnaire. Every time he answers a question, a temporary answer is saved for UX reasons.
    - Supplier can partially answer the questionnaire and finish it later
3. The question replies are stored and go through analysis, where EPES operators and vendors can view their assessment analysis results
    - During the answering process and after the questionnaire has been committed, the pilot partner can utilize visual graphs to view his progress
    - When the questionnaire is finished, a report can be generated for this questionnaire. This can be a pdf report, an html page, or a shared link
4. EPES operators additionally view the vendors' assessment results, as produced by the analysis
    - Vendors can view their own results, as well as other vendors' results as benchmarking

Set Vendor Questionnaires

The Self-assessment tool has one additional feature available only to the EPES operators: the capability to define and enable specific sets of questionnaires for each vendor. For example, a vendor that provides software solutions will a have different set of questions from one that provides supporting services.

Internal architecture

Self-assessment tool, with the components depicted in figure 25, is a web-based software tool architected to provide the following capabilities:

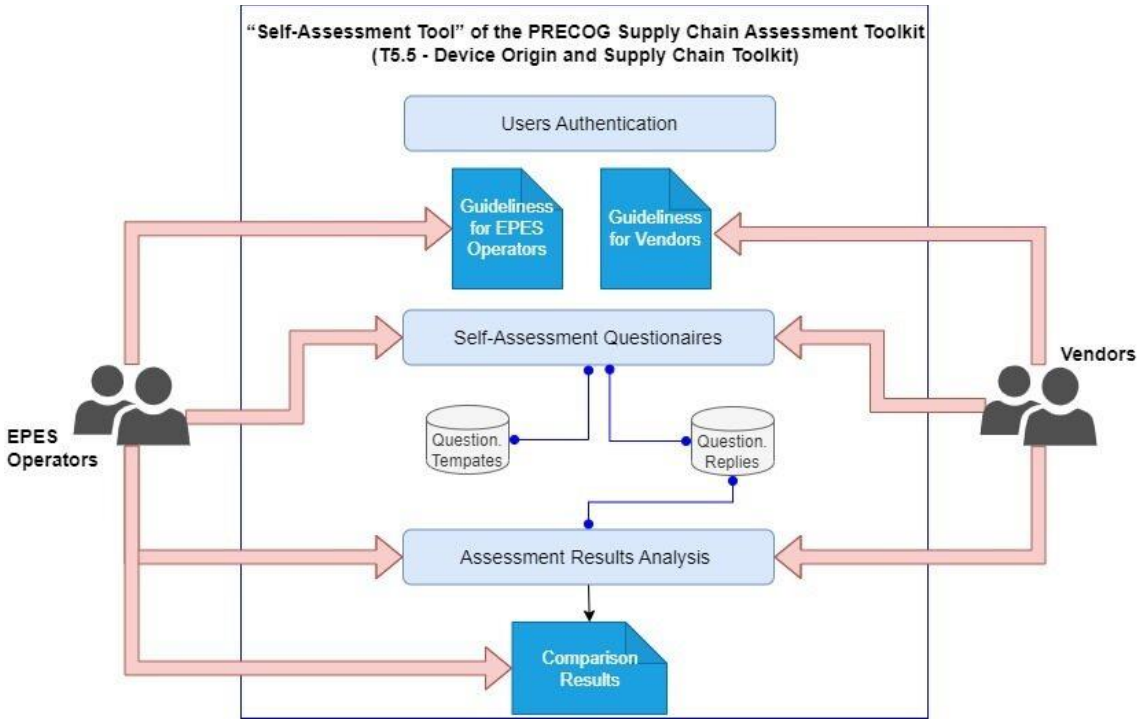

Figure 25 – Self-assessment tool Architecture Components

- **User authentication**

Authentication is one of the fundamental pillars in cyber-security since it verifies a user or device before allowing access to a system or resource.

Access to the Self-assessment tool software capabilities will be authenticated. The users can be created and maintained on the tool itself, and/or they can exist on an EPES directory or IDP. The integration of the tool with the external directories or IDPs is based on well-known protocols like LDAPS and SAML.

- **Guidelines and standards**

Guidelines and standards are references to well-known and well acceptable standards and guidelines regarding the supply chain. Vendors and EPES may share common standards but also may have differences. More details regarding guidelines and standards are available on the "Description of Components" section

- **Self-assessment**

Self-assessment is a process by which an organization evaluates its own cyber-security posture. This can be done by using a variety of tools and techniques, such as questionnaires, checklists, and interviews. The goal of self-assessment is to identify areas where the organization can improve its cyber-security practices.

Self-assessment can help organizations to:

- Identify potential vulnerabilities and risks in their cyber-security posture
- Assess the effectiveness of their existing cyber-security controls.
- Prioritize areas for improvement

- Develop and implement a plan to improve their cyber-security posture

The R^2D^2 Self-assessment tool is a comprehensive tool that can be used to evaluate the cyber-security posture of an organization's supply chain. The tool covers a wide range of topics, including:

- Supply chain management practices
- Cyber-security practices of vendors
- Use of IoT devices and cloud solutions

Some additional benefits of self-assessment are:

- It can help organizations to comply with regulations
- It can improve the organization's risk management process
- It can help to build a culture of security within the organization
- It can help to improve the organization's reputation

The tool uses a rating scale or scoring system to quantify the cyber-security maturity in each area. This helps organizations to identify areas where they need to improve their cyber-security practices.

The Self-assessment tool is regularly updated to stay ahead of evolving cyber-security threats. This ensures that organizations are always using the most up-to-date information to evaluate their cyber-security posture. Description of components is as follows.

Guidelines for Operators

A guidelines component provides additional reference materials, guidelines, best practices, or educational resources related to supply chain security management. It offers users supplementary information to enhance their knowledge and capabilities.

Guidelines for Vendors

Standards for energy vendors in the EU are a set of rules and guidelines that govern the development, testing, and deployment of software for use in the energy sector. They are designed to ensure the safety, security, and efficiency of energy software. The following table (Table 6) depicts an indicative list of guidelines.

Table 6 – Supply Chain guidelines

EPES guidelines	VENDOR guidelines
ENISA Good Practices for Supply Chain cyber-security [10]	ENISA Good Practices for Supply Chain cyber-security [10]
NIST Cybersecurity Framework (CSF) [11]	NIST Cybersecurity Framework (CSF) [11]
ISO/IEC 27001:2022 - Information security management systems [12]	ISO/IEC 27001: 2022 - Information security management systems [12]
Cyber-security Maturity Model Certification (CMMC) [13]	Cyber-security Maturity Model Certification (CMMC) [13]
CERT Resilience Management Model (CERT-RMM) [14]	NIST Special Publication 1800-34 “Validating the Integrity of Computing Devices” [15]
MITRE System of Trust (CERT-RMM) [16]	MITRE System of Trust (CERT-RMM) [16]

NIST SP 800-161 Rev. 1 - cyber-security Supply Chain Risk Management Practices for Systems and Organizations [17]	NIST Software Supply Chain Security Guidance [18]

Self-assessment questionnaires

This component executes the assessment questionnaire and presents the questions to users in a structured manner. It handles the logic and flow of the questions, ensuring that users progress through the assessment correctly. Additionally, this component captures and stores user responses to the assessment questions. It validates and stores the data securely for further analysis and reporting.

Questionnaires Database

The questionnaires in the Self-assessment tool are preset. There are different sets of questions, depending on the role of the user being assessed, EPES or vendor. When accessing the Self-assessment tool, the user is required to select the appropriate set of questions based on their role, which they subsequently can answer, save for later, resume and finally submit. Their replies are saved in their appropriate DB (table).

Questionnaires Replies DataBase

The EPES operators and vendors complete the corresponding questionnaires in the Self-assessment tool. After submitting their final responses, the replies are stored in another dedicated database (table), so that they can be analysed in the next step of the assessment. The replies cannot be accessed by users before they are put through analysis.

Assessment results analysis

This component captures and stores user responses to the assessment questions. It validates and stores the data securely for further analysis and reporting. The tool includes a scoring and evaluation component. It assesses the user's responses against predefined criteria or benchmarks to generate scores, rankings, or ratings. This component helps identify areas of strength and areas that require improvement.

Results Reporting

All users can view the results of the analysis of their replies. Additionally, all vendors can also view the results of other vendors of the industry, and compare their results with the overall results in the industry. EPES operators can view not only the results of other operators, in the same manner, but also those of the vendors

Techniques & Algorithms

The elf-assessment tool is built upon the latest technology standards in order to provide a modern and secure technological environment.

- TLS 1.2 protocol to secure data in transit
- If an external directory or IDP is used, SAML over HTTPS or NTLM, or LDAPS is used
- The algorithm CYBER uses to score the answers on the questionnaire will be the following:
- Sum(Average(answers) * domain weight) * maximum possible weighted score / 100
- The scores of each questionnaire are kept temporarily as session data and are committed to database once the user submits the questionnaire

Data Exchanges & Interfaces

The Self-assessment tool is a rather autonomous tool. The data is input from users. In detail:

- Pilot site practices: This will be the input to the tool regarding current pilot site practices, based on which the provided tool will perform the required assessment
- Vendor practices: This will be the input to the tool regarding current vendor practices, based on which the provided tool will perform the required assessment
- Self-assessment results: The results of the self-assessment performed by the vendors and pilot site, against the guidelines developed by T5.5 partners.

User interface

The Self-assessment tool will be a web-based environment. The following mock-up (shown in Figure 26) depicts the Self-assessment tool log-in page.

Landing Page: The landing page introduces the purpose of the tool and provides a brief overview of its benefits. It also includes a login or registration option for users to create an account or access their existing accounts.

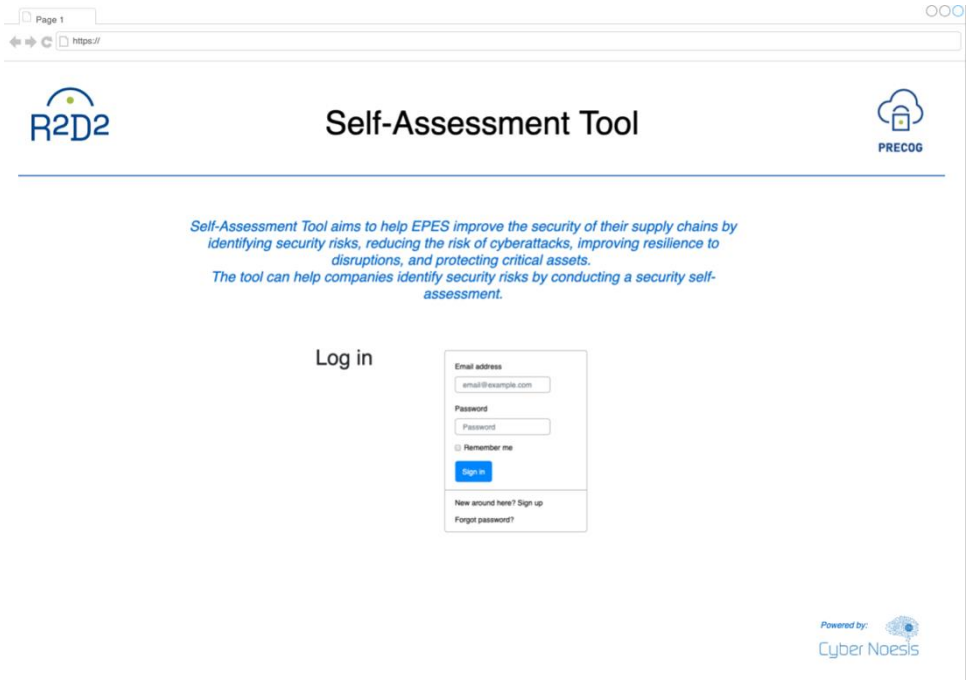

Figure 26 – Self-assessment tool users' authentication

- **Dashboard**: Upon logging in, users are directed to a dashboard (shown in Figure 27) that serves as the main hub of the application. The dashboard provides an overview of the user's assessment progress, completed assessments, and any notifications or recommendations based on previous assessments.

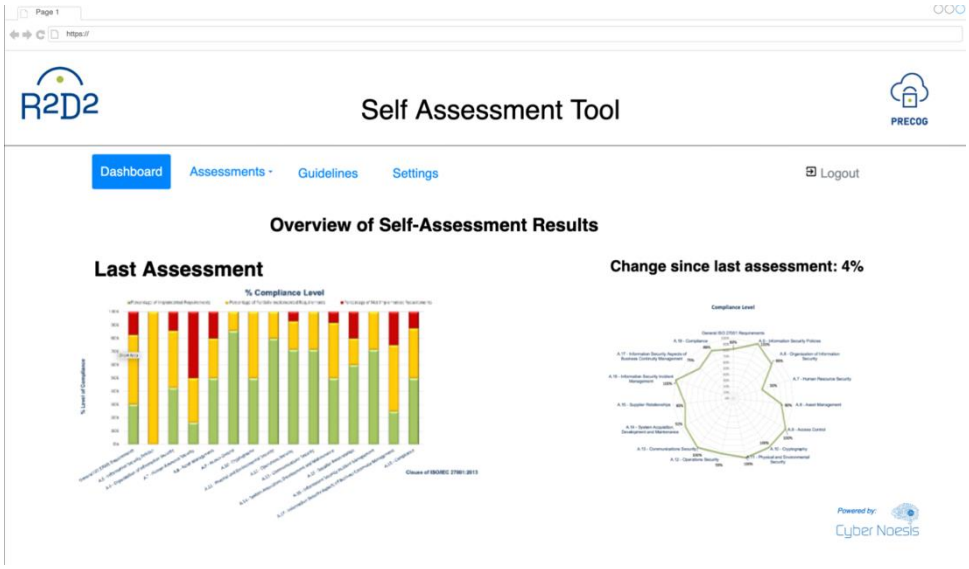

Figure 27 – Self-assessment tool results dashboard

- **Assessment Selection**: The dashboard (shown in Figure 28) includes a section where users can select the specific self-assessment related to supply chain security management they want to undertake. This could include assessments focused on risk assessment, compliance, supplier evaluation, or other relevant areas.

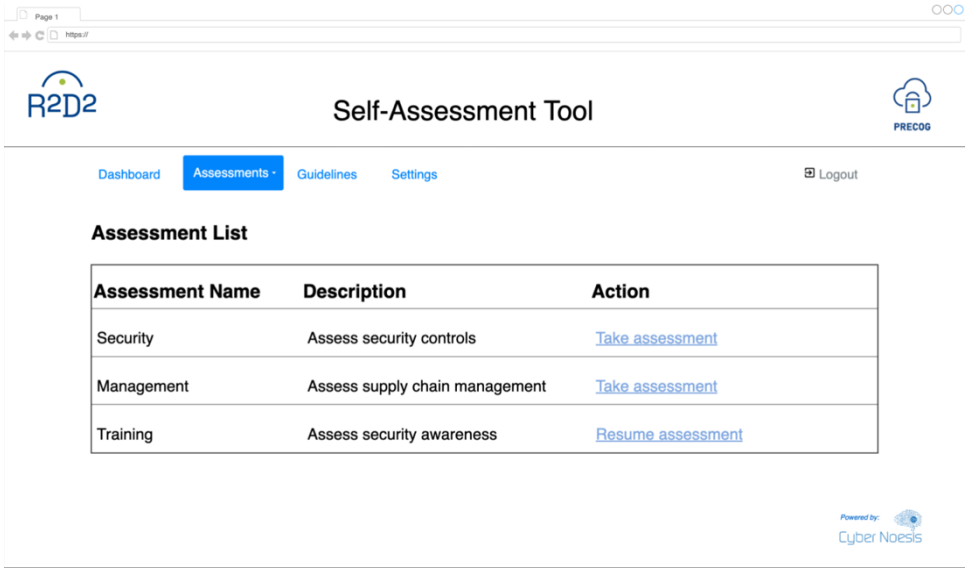

Figure 28 – Self-assessment tool List of available assessments

- **Questionnaire Interface**: The questionnaire interface (shown in Figure 29) presents a series of questions related to the chosen assessment category. The questions could be displayed one at a time or grouped in sections, depending on the complexity and length of the assessment. Each question should be clear, concise, and accompanied by any necessary instructions or clarifications.

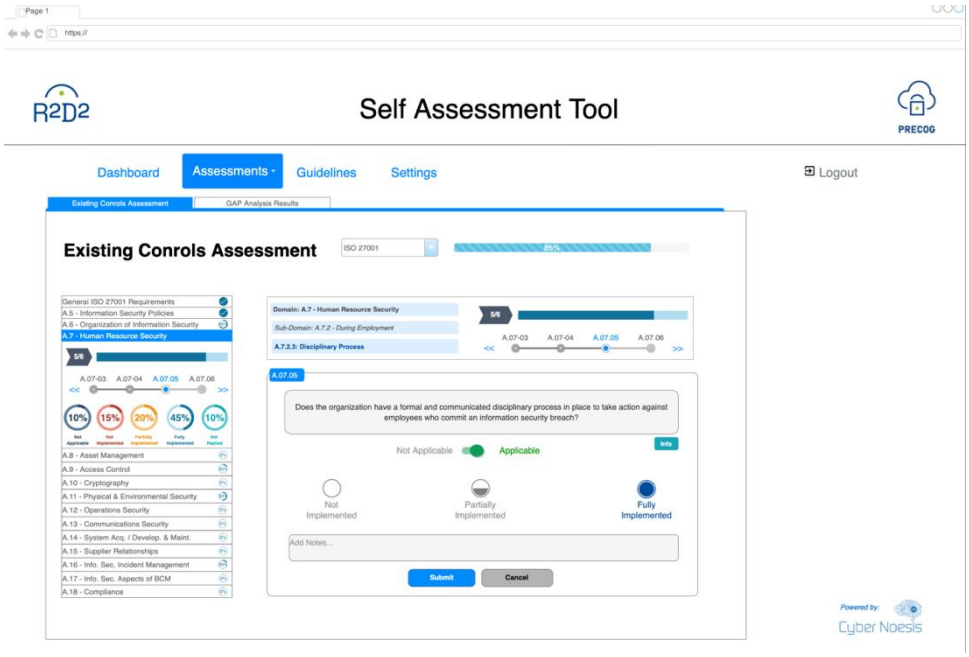

Figure 29 – Self-Assessment Tool questionnaire

- **Response Input**: Users would provide their responses to each question using various input methods. Common input types could include multiple-choice options, checkboxes, dropdown menus, or text fields for open responses. The interface should make it easy for users to select or input their responses accurately.
- **Progress Tracking**: The UI features a visual indicator of the user's progress within the assessment, such as a progress bar or percentage completion display. This helps users understand how far they are into the assessment and how much remains.
- **Save and Resume**: Users have the option to save their progress and resume the assessment at a later time if needed. This feature ensures flexibility and convenience, allowing users to complete the assessment at their own pace.
- **Results and Reports**: Once the assessment is completed, the UI can generate comprehensive reports or summaries based on the user's responses. The reports may include visualizations, graphs, or charts to present the assessment results clearly. It could highlight areas of strength, weaknesses, and recommendations for improvement
- **Guidelines**: The UI provides supplementary resources, such as best practice guidelines, documentation, and links to external sources, to support users in addressing identified gaps or improving their supply chain security practices.
- **Account Management**: The UI includes options for users to manage their accounts, including updating profile information, changing passwords, and accessing past assessment reports
- **Help and Support**: A dedicated section for help and support is available, providing users access to user guides, or a contact form to seek assistance if they encounter any issues or have questions.

Resources

The Self-assessment tool uses the following technology stack:

- Servers
  - A Virtual Machine will host the application server and the web server of the Sandbox tool
  - A Virtual Machine will host the DB to store
    - the Supply Chain standards and guidelines,
    - the questionnaires,
    - the answers on the questionnaires
    - the analysis results,
- Operating System
  - The operating systems will be Linux flavour
- Web server
  - The web server is an Apache HTTP Server 2.4.57
- Database management system
  - The Database management system is MySQL Community Server 8.0.34
- Development language
  - Python and HTML5 will be used
- Software Libraries
  - Python Django and Bootstrap css

- Deployment model

The deployment model describes where the solution will be installed. It may be on a physical server(s), on a Virtual Machine(s) or on a public cloud. The Self-assessment tool will be installed on a Virtual Machine infrastructure.

The database that is installed on a VM hosted on CYBER's infrastructure and stores:

- the Supply Chain standards and guidelines,
- the questionnaires,
- the answers on the questionnaires
- the analysis results,
- is installed on a VM hosted on CYBER's infrastructure.

The application server and the web server are also installed on VM hosted on CYBER's infrastructure.

- Operating System – The operating system is CentOS Linux
- Web server – The web server is an Apache HTTP Server
- Database management system – The Database management system is MySQL Community Server
- Development language – CYBER uses Python and HTML5
- Software Libraries – Python Django and Bootstrap css

4.6 IMPLEMENTATION AND DEPLOYMENT PLAN

This section covers the general WP5 implementation and development plan for the second iteration (table 7 on page 77), which is mandatory for all WP5 tasks to follow and shall be

adjusted according to each specific tool implementation and deployment specifically based on chosen development language, architecture and any other peculiarities.

Activity 1: Background service and functionality development, including unit and integration tests.

In this step, input from the first iteration is analysed and combined with new data produced by pilots and tool providers. Based on analysis, the practical start of the second iteration will be made. It includes development tasks for one or more use cases and pilots and it includes development tasks which are use case specific. In this phase, the tool providers will plan their tasks in a most effective way to support best development results.

Activity 2: User Interface development (T5.6)

In this step, preliminary sketches and ideas (delivered in this document) of WP5 user interfaces are analysed as input for T5.6, where user interfaces are planned, designed and made. In this activity it is decided which tools need GUI and which tools are used via other interfaces. At the end of phase two, sufficient demo UI is ready for deployment on the pilot's environments for every tool that needs it.

Activity 3: Continuous SW quality and dependency.

This activity is intended to be running in parallel with Activity 1 and 2. It is a good and endorsed practice to have quality and security checks running on code to find any issues within coding or dependencies, as it is humanly impossible to keep track of all issues and vulnerabilities. Following scans and tools, or other equivalent, are recommended:

- For quality checks a SonarQube or similar
- Dependency scanning for security vulnerabilities: MendBolt / Dependency checker / Coverity
    - Libraries used and potential weaknesses (scoring CVVS)
    - Depending on the chosen deployment, additional checks could be envisaged, e.g. securitization of docker images (https://docs.docker.com/docker-hub/vulnerability-scanning/

Activity 4: Delivery of preliminary technical result which cover main functionality of tools (MS5)

In this step, it will be determined the level of details of demos which will be conducted to deliver milestone MS5 (Preliminary technical results). Based on the plan, preliminary technical results will be delivered.

Activity 5: Acceptance testing to meet all acceptance criteria and requirements

In this step, previously defined use cases, requirements and general KPIs are analysed. Based on analysis, details of testing methods and approaches are selected and conducted to meet the acceptance criteria.

Activity 6: Early SW delivery in pilots environment

In this activity tools are packaged and delivered to the pilot`s environments providing functionality defined in use cases.

Activity 7: SW documentation & release deliverable preparation

Work conducted from throughout activities 1 to 6 are documented and is prepared for the final release deliverable.

Activity 8: Final SW release delivery (MS6)

In this step WP5 tool providers (GUARD, S2, ELPROS and CYBER), in collaboration with pilots sites, will deliver WP5 tools (KSI tool, Tokenization tool, CARMEN tool, UniFusion platform, Sandbox tool and Self-assessment tool), which fulfil R^2D^2 Specific objectives SO1 (To contribute to the improvement of the overall security and resiliency in power system) and SO3 (To increase the cyber-security and cyber-resilience in OT and IT of the EPES) alongside the necessary documentation on how to set up, run and use the tools. Conducted work will also be input for WP7 (Integration, Demonstration and Validation) M19 – M36 [1].

Table 7 – Deployment plan of second iteration

Months	M13 - Oct.23				M14 - Nov.23					M15 - Dec.23				M16 - Jan.24				M17 - Feb.24					M18 - Mar.24				M19 - Apr.24				M20 - May24					M21 - Jun.24				M22 - Jul.24				M23 - Aug.24					M24 - Sep.24			
Weeks	40	41	42	43	44	45	46	47	48	49	50	51	52	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39
Activity 1																																																					
Activity 2																																																					
Activity 3																																																				
Activity 4																																																				
Activity 5																																																				
Activity 6																																																				
Activity 7																																																				
Activity 8																																																				

5 Conclusions and next steps

This document presents the contribution from WP5, which is dedicated to Prevention Systems For Energy Infrastructures Security. Work in WP5 started in parallel with WP2, which defined use cases and requirements. Based on 10 defined use cases, this document describes six tools which will be practically implemented in the second iteration (after delivery of this document) of the R^2D^2 project (M13 – M24). Tools are developed under a product called PRECOG which is one of four products delivered in the R^2D^2 project. List of tools provides innovative approach for four pilots and is as follows:

- **KSI tool** – developed and implemented by GUARD, tested in two use cases. Tool signs the data and then provides integrity validation for signed data and thereby increases trust for signed data.
- **Tokenization tool** – developed and implemented by GUARD, tested in four use cases. Tool tokenizes data and through immutability verification increases trust for the data which has been tokenized by Tokenization tool.
- **CARMEN tool** – developed and implemented by S2, tested in four use cases. Using port mirroring and port forwarding through a secure interface, the tool improves LW network observability and improves system security. In addition, the tool detects anomalies associated with cyber-security thanks to the traffic characterization at control, operation and supervision levels. The tool also detects patterns, correlated with already happened and documented cyber-attacks, to detect new potential threats in order to attribute and classify them.
- **Sandbox tool** – developed and implemented by CYBER, tested on one use case. The tool monitors and verifies the security of newly deployed components in an EPES staging environment. Sandbox not only integrates the capabilities of the partner`s solutions (CARMEN, Claudia, List, Argos, KSI tool) but further improves the process to evaluate the security status of a new piece of software before integrating it in the production environment.
- **Upgraded UniFusion platform** – developed and implemented by ELPROS will be tested in three use cases. The platform as a complex communication system collects data from various data sources. The platform will provide secure communication protocols using the latest protective communication measures.
- **Self-assessment tool** – developed and implemented by CYBER, tested in one use case. The tool constitutes a central point of reference with the widely accepted Supply Chain security standards and guidelines. Additionally provides assessments (questionnaires) for the EPES and their Vendors to help them identify their current security posture against the standards, evaluate risks and help them to improve the security of their supply chain practices.

In the second iteration of R^2D^2 project WP5 participants will turn theoretically collected experience into six practical tools implemented and tested in pilot`s environments. WP5 general implementation plan will be followed and tailored based on pilot partners and tool provider needs to enhance cyber-security of EPES digital and physical assets. Work conducted in the second iteration will fulfil milestone MS6 (Final technical results) and also will serve as input for WP7 (Integration, Demonstration and Validation) M19 – M36.

6 References

[1] R^2D^2 WP7 "Integration, Demonstration and Validation"
[2] "R2D2 Grant Agreement. EC, 2022"
[3] R^2D^2 "Requirements and Detailed Architecture Design"
[4] ISO/IEC 27036-1:2021 Cybersecurity — Supplier relationships — Part 1: Overview and concepts https://www.iso.org/standard/82905.html
[5] ISO/IEC 20243-1:2018 Information technology — Open Trusted Technology ProviderTM Standard (O-TTPS) — Mitigating maliciously tainted and counterfeit products — Part 1: Requirements and recommendations https://www.iso.org/standard/74399.html
[6] FERC, Cyber and Grid Security https://www.ferc.gov/industries-data/electric/industry-activities/cyber-and-grid-security
[7] CARMEN https://s2grupo.es/en/herramientas/carmen/
[8] MITRE https://attack.mitre.org/
[9] MISP https://www.misp-project.org/
[10] ENISA Good Practices for Supply Chain cyber-security (https://www.enisa.europa.eu/publications/good-practices-for-supply-chain-cybersecurity)
[11] NIST Cybersecurity Framework (CSF) - https://www.nist.gov/cyberframework
[12] ISO/IEC 27001:2022 - Information security management systems, https://www.iso.org/standard/27001
[13] Cyber-security Maturity Model Certification (CMMC) - https://www.cisa.gov/resources-tools/resources/cybersecurity-maturity-model-certification-20-program
[14] CERT Resilience Management Model (CERT-RMM) - https://insights.sei.cmu.edu/library/cert-resilience-management-model-cert-rmm-version-12/
[15] NIST Special Publication 1800-34 "Validating the Integrity of Computing Devices" - https://csrc.nist.gov/pubs/sp/1800/34/final
[16] MITRE System of Trust (CERT-RMM) - https://sot.mitre.org/framework/system_of_trust.html
[17] NIST SP 800-161 Rev. 1 - Cybersecurity Supply Chain Risk Management Practices for Systems and Organizations - https://csrc.nist.gov/pubs/sp/800/161/r1/final
[18] NIST Software Supply Chain Security Guidance - https://www.nist.gov/itl/executive-order-14028-improving-nations-cybersecurity/software-supply-chain-security-guidance

8. ANNEX I

Sandbox Workflow Usage Examples

a. An **EPES_A user wants to install a software update** which has not been evaluated yet.

1. The EPES_A user logs in the Sandbox tool and searches if the specific software has been tested by someone else.
2. The software update has not been evaluated yet by someone else.
3. An EPES_A user downloads a software update.
4. The EPES_A user enters the new software data in the Sandbox tool (name, version, date, etc) - but not the software itself.
5. The EPES_A user uses the Sandbox tool to upload the new software package and calculate the software hash digest. The hashing process is integrated in the Sandbox tool.
6. The software is installed on the selected equipment (to be deployed in the staging environment)- eg a SLAM (it is an upgrade).
7. The equipment is deployed in the staging environment
8. The EPES_A starts evaluating the equipment's new software utilizing the Sandbox tool. This is achieved by monitoring the equipment's communication (using the S2 tools).
9. The Sandbox tool - using the S2 tools - proposes an assessment result.
10. A security analyst, using the Sandbox tool, reviews the results, classifies the software and signs the hash digest calling the web APIs - through the Sandbox tool - GUARD has exposed to CYBER.
11. The results, the signed hash, the proofs, and the software description are stored in Sandbox tool DB.

b. An EPES_B user wants to install a software update which has been already evaluated

1. An EPES_B user logs in the Sandbox tool and searches if the specific software has been tested by someone else.
2. The software update has already been tested by EPES_A.
3. An EPES_B user downloads the software update from the Vendor's site.
4. An EPES_B user uploads the software update to the Sandbox tool to calculate the software hash digest.
5. Sandbox tool compares the two hash digests (the one signed and the new one). If the two digests are equal, the Sandbox tool calls GUARD's APIs to validate the integrity of the signed hash.

EPES_B gets the validation results.

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union. Neither the European Union nor the granting authority can be held responsible for them. Horizon Europe Grant agreement N° 101075714.