ZEROSWARM

ZERO-enabling Smart networked control framework for Agile cyber physical production systems of systems

D5.5 - Anomaly detection & countermeasure selection tools.R1

Project funded by Horizon Europe, Grant Agreement #101057083

Topic	HORIZON-CL4-2021-TWIN-TRANSITION-01-08
Project Title	ZERO-enabling Smart networked control framework for Agile cyber physical production systems of systems
Project Number	101057083
Project Acronym	Zero-SWARM
Contractual Delivery Date	M17
Actual Delivery Date	M18
Contributing WP	WP5
Project Start Date	01/06/2022
Project Duration	30 Months
Dissemination Level	Public
Editor	CERTH
Contributors	S21Sec, NX-SE

Authors List

Leading Author (Editor)			
Surname	Initials	Beneficiary Name	Contact email
Mpatziakas	AM	CERTH	ampatziakas@iti.gr
Co-authors (in alphabetic order)			
Surname	Initials	Beneficiary Name	Contact email
Borne	RB	S21Sec	rborne@s21sec.com
Egaña	JE	S21Sec	jegana@s21sec.com
Hatzidiamantis	NH	CERTH	*hatzidiamantis@iti.gr*
Lazaridis	GL	CERTH	glazaridis@iti.gr
López	OL	S21Sec	olopez@s21sec.com
Mpatziakas	AM	CERTH	ampatziakas@iti.gr
Deshmukh	SD	NX-SE	shreya.deshmukh@se.com
Fritz	AF	NX-SE	artur.fritz@se.com

Reviewers List

List of reviewers (in alphabetic order)			
Surname	Initials	Beneficiary Name	Contact email
Guerra	RG	NEU	rolando.guerra@neutroon.com
Meeßen	MM	CCI	m.meessen@connectedindustry.net
Khodashenas	*PK*	*HWE*	*pouria.khodashenas@huawei.com*

Document History

Document History		
Version	Date	Remarks
0.1	01/9/2023	Table of Content created
0.2	30/10/2023	Input from CERTH and S21 (sections 2,3,4)
0.3	7/11/2023	Additional material in sections 2,3,4 from all contribution
0.5	13/11/2023	Final contributions from all, document ready for internal review
0.9	16/11/2023	Internal review comments addressed
1.0	21/11/2023	Final document (editorial changes)

DISCLAIMER OF WARRANTIES

This document has been prepared by Zero-SWARM project partners as an account of work carried out within the framework of the contract no 101057083.

Neither Project Coordinator, nor any signatory party of Zero-SWARM Project Consortium Agreement, nor any person acting on behalf of any of them:

- makes any warranty or representation whatsoever, express, or implied,
    - with respect to the use of any information, apparatus, method, process, or similar item disclosed in this document, including merchantability and fitness for a particular purpose, or
    - that such use does not infringe on or interfere with privately owned rights, including any party's intellectual property, or
- that this document is suitable to any user's circumstance; or
- assumes responsibility for any damages or other liability whatsoever (including any consequential damages, even if Project Coordinator or any representative of a signatory party of the Zero-SWARM Project Consortium Agreement, has been advised of the possibility of such damages) resulting from your selection or use of this document or any information, apparatus, method, process, or similar item disclosed in this document.

Zero-SWARM has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No 101057083. The content of this deliverable does not reflect the official opinion of the European Union. Responsibility for the information and views expressed in the deliverable lies entirely with the author(s).

Executive Summary

Deliverable D5.5 “Anomaly detection and countermeasure selection tools.R1”, is the first technical deliverable produced by T5.5. The Zero-SWARM project aims to deliver multiple innovations to be used by the European manufacturing sector in the context of the so-called Industry 4.0 (I4.0). I4.0 can be defined as “the current trend of automation and data exchange in manufacturing technologies, including cyber-physical systems, Industrial Internet of Things, cloud computing and cognitive computing to create the “smart factory”. Industries are evolving by connecting their infrastructures to IT technologies with the aim of boosting their potential and creating new value. We witness an accelerated interconnection of elements that have not been designed with robust security aspects, and potentially their exposure to the internet. This leads the exposition of the industrial domain to several threats and risks. D5.5 focuses on the cybersecurity aspects of Cyber Physical Systems of Systems and Industrial Internet of Things, introducing anomaly detection and mitigation mechanisms as a means of securing such industrial systems and addressing some of the emerging risks of I4.0.

The document initially connects these cybersecurity mechanisms with the rest of the Zero-SWARM project: the general architecture as presented in D2.2, the Zero-Swarm cybersecurity reference architecture presented in D2.3 and the penetration testing mechanisms presented in D5.4. Then section 2 presents a brief description of State of the Art and common practise approaches for anomaly detection, classification, and response in 5G and IIOT networks along with a short reference to the related standards. Section 3 presents the mechanisms developed in the project, namely: first a module that handles Security Information and Event Management (SIEM) along with a Security Orchestration, Automation and Response (SOAR). Then a module that utilizes multiple state of the Art AI algorithms to enable detecting, classifying and counter-measuring anomalies based on network traffic. Finally, section 4 focuses on the presentation of details concerning our plans for integration and validation efforts. The modules introduced in section 3 will be integrated in the IEC 61499 platform created in T5.1 and detailed in D5.1 “Distributed automation and information management”. A high-level architecture for the purposes of this integration is presented. Additionally, a setup used for lab testing of the SIEM/SOAR modules along with the tool that will used for network traffic monitoring and capture is presented.

Final development and validation results of the presented modules, along with the specific scenarios that will be used to validate these modules in the project trials, will be reported in the next version of this deliverable, namely D5.10 “Anomaly detection and countermeasure selection tools.R2”.

Table of Contents

List of Figures

List of Tables

List of Acronyms

Abbreviation	Description
API	Application Programming Interface
ACM	Anomaly Classification module
ADM	Anomaly detection Module
AI	Artificial Intelligence
APT	Advanced Persistent Threats
B5G	Beyond 5G
CNN	Convolutional Neural Networks
CPS	Cyber Physical System
CPSoS	Cyber Physical Systems of Systems
CSM	Countermeasure Selection Module

DBN	Deep Belief Networks
DCCNN	Dilated Causal Convolutional Neural Network
DDoS	Distributed denial
DL	Deep Learning
DNN	Deep Neural Network
DoS	Denial of Service
dPAC	Distributed Programmable Automation Controller
DPI	Deep Packet Inspection
EDR	Endpoint Detection and Response
HIDS	Host Intrusion Detection Systems
I/O	Input and Output
IAT	Inter-Arrival Time
IEC	International Electrotechnical Commission
IIOT	Industrial Internet of Things
IIRA	Industrial Internet Reference Architecture
IOT	Internet of Things
IRS	Intrusion Response System
IT	Information Technology
KPI	Key Performance Indicator
LAN	Local Area Network
LSTM	Long Short-Term Memory Recurrent Network
MitM	Man-in-the-Middle
ML	Machine learning
MO	Multi-objective
MQTT	Message Queuing Telemetry Transport
NCC	normalized normal constrained
NIDS	Network Intrusion Detection Systems
NN	Neural Network
OPC-UA	Open Platform Communications Unified Architecture
OT	Operational Technology
RAN	Radio Access Networks
RNN	Recurrent Neural Network
ROC	receiver operating characteristic curve
SAE	Stacked AutoEncoders
SCADA	Supervisory Control And Data Acquisition
SDN	Software Defined Networking
SEM	security event management
SIEM	Security information and event management
SIM	security information management
SOAR	Security Orchestration, Automation and Response
SOC	Security Operations Center
SoTA	State-of-the-art
TCP	Transmission Control Protocol
Tx.x	Task x.x
UAD	Ultralight Anomaly Detection
UC	User Control
VAE	Variational Autoencoder
VPN	Virtual Private Network
WLAN	Wireless Local Area Network

WP	Work Package

1 Introduction

Industry and Operational Technology (OT) environments have traditionally remained segregated from the Information Technology (IT) sphere. Nevertheless, industrial systems are now in a state of evolution, as they increasingly integrate their infrastructures with IT technologies to boost efficiency and productivity. This rapid convergence, where multiple underlying technologies are been used in novel and innovative contexts, is resulting in the interconnection of components lacking robust security measures. This consequently leads to exposing production and manufacturing infrastructures and processes to the IT and Internet realms. This, in turn, exposes the industrial domain to a multitude of threats and risks.

Anomaly detection and mitigation mechanisms against cyber-attacks in industrial systems, that utilize multiple technologies and approaches simultaneously such as 5G/B5G, CPSoS and IIOT, are critical components for safeguarding the integrity and security of these infrastructures and processes. With the increasing connectivity of industrial systems and the rise of sophisticated cyber threats, the need for robust defences is paramount. Modern anomaly detection methods utilize advanced algorithms to continuously monitor network traffic, system behaviour, and data patterns, pinpointing any unusual activities that might indicate a cyber-intrusion. Once an anomaly is identified, immediate mitigation measures can be applied, such as isolating compromised systems, updating security protocols, or even shutting down critical components if necessary to prevent potential damage or data breaches. In an era where industrial systems are prime targets for cyber-attacks, anomaly detection and mitigation mechanisms serve as an essential frontline defence, ensuring the resilience and reliability of these essential systems.

1.1 Purpose of the document

D5.5 - “Anomaly detection and countermeasure selection tools.R1” is the first deliverable of T5.5. This task, which started on M7, aims to create mechanisms that facilitate the detection and mitigation of threats and security related events against the CPSoS. In the current document we present the state of the Art (SotA) along with common practices about anomaly detection and mitigation/countermeasure action selection in the context of industrial environments along with relevant information about different attacks. Then, the mechanisms developed for the project are presented in detail. Finally, we present our plans for integration with the IEC 61499 platform created in T5.1 and which will be utilized to validate the developed mechanisms.

The next and last version of this document will contain the final descriptions of the proposed mechanisms along with results for their evaluation and validation. It will be named D5.10 “Anomaly detection & countermeasure selection tools.R2” and is due in M24.

1.2 Structure of the document

The following subsection contains the structure of the document:

- **Chapter 1** is an introduction to the whole document, describing its scope and purpose, provides the connection of deliverable D5.5 to other deliverables of the project; its structure, the delivery plan during the project’s lifetime, as well as outlining the task’s objectives.
- **Chapter 2** provides the State of the Art and Common Practices of Anomaly detection and countermeasure methods focused on IIoT and CPSoS
- **Chapter 3** presents the modules developed on Zero-SWARM for efficient Anomaly / Intrusion Detection and the robust countering.

- **Chapter 4** outlines the plans for integration of the T5.5 modules in an IEC 61499 platform simulation to evaluate and verify the functionalities of the modules developed in T5.5 in a realistic manner before deploying and testing them in the Zero-SWARM trials.
- **Chapter 5** will conclude the work of this deliverable, provide some insight regarding the next steps of this task, and will comment on the task's activities.

1.3 Connection with the relative tasks and deliverables

Deliverable D5.5 "Anomaly detection and countermeasure selection tools.R1" is the first technical cybersecurity document of a series of four deliverables, in the scope of task T5.5. Due to the nature of this task, the cybersecurity activities are horizontally placed within the project, covering the cybersecurity aspects of nearly the entire project. For this reason, D5.5 is directly connected with a series of other deliverables of the project. More specifically, there is a connection among D5.5 and D2.2 [1], D2.3 [2], D5.1 [3], D5.4. [4], D6.1 [5] and D6.2 [6]. The following section briefly presents the interconnection between these deliverables.

D2.2 contains the project architecture while D2.3 contains the projects' Cyber-security implementation templates, the methodological approach chosen and a reference cybersecurity architecture. The mechanisms presented in this deliverable consider the architectural choices and follow the methodology proposed in the WP2 deliverables. A mapping of the modules developed in T5.5 and presented in this deliverable to the CPSoS deployment view / integration with responding IEC 62443 reference levels defined in D2.2 and the relevant Cybersecurity layers defined in D2.3 is presented in Figure 1.

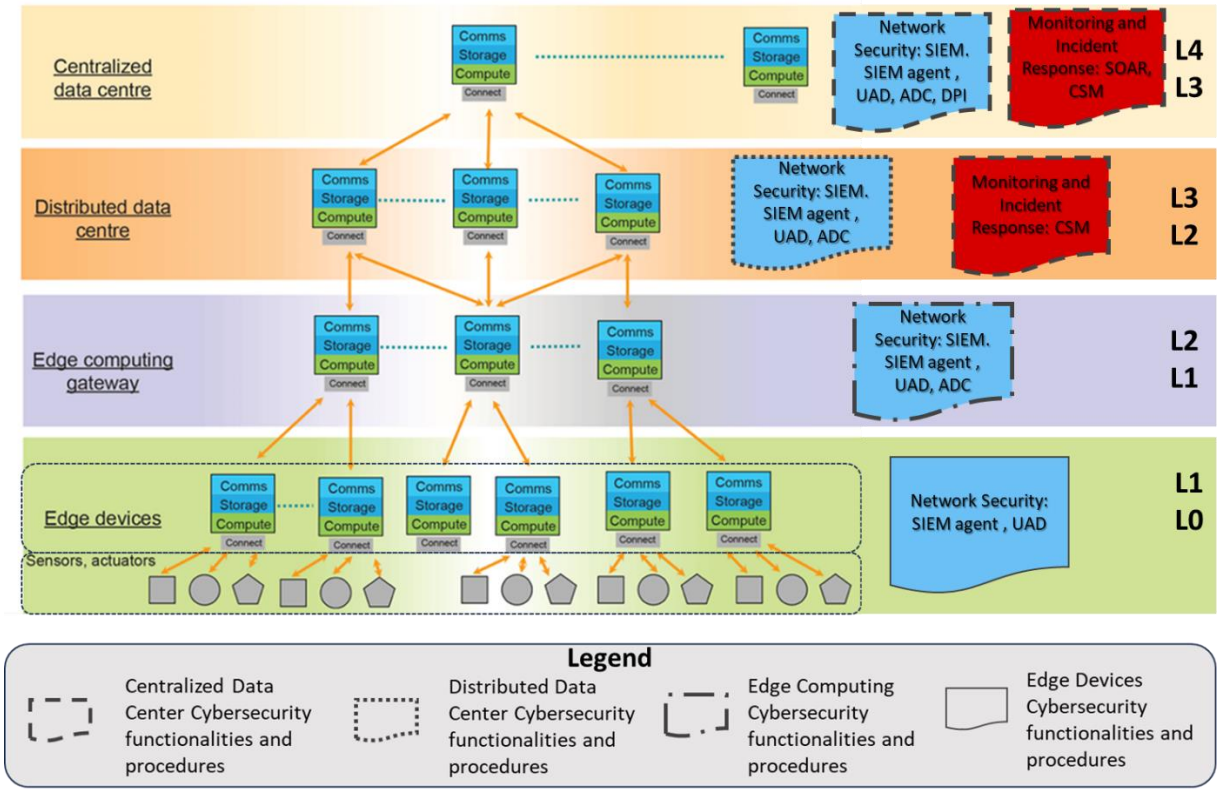


Figure 1 CPSoS deployment view / integration with responding IEC 62443 reference levels Cybersecurity layers transversal to CPSoS deployment view

D5.1 contains information for the distributed automation and information management approach that will be utilized in the project and the mechanisms of D5.5 will need to follow to easily integrate with the trials.

D5.4 presents two modules developed in the project, one named Penetration Testing module that produces inputs utilized by the anomaly detection and countermeasure selection modules presented in section 3.2 and one named Hypothesis testing module that receives input from the same D5.5 modules. The process is depicted in Figure 2.

Finally, the interfaces that will be used by the modules presented in D5.5 were defined and described in D6.1, while functional tests and related KPI for the modules were presented in D6.2.

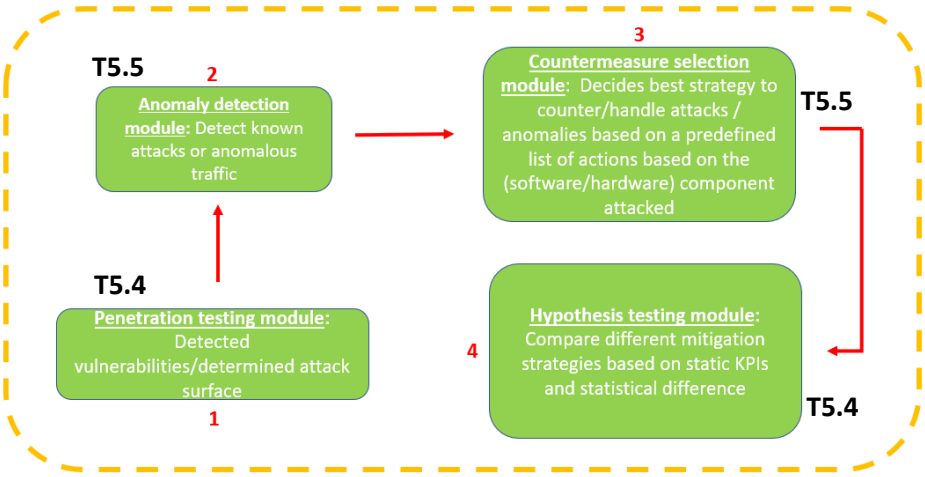


Figure 2: Unified high-level T5.4 - T5.5 interconnections

2 State of The Art & Common Practices

This purpose of this section is to briefly present the background knowledge utilized towards the design, planning and development of the modules presented in section 3. It contains the SoTA and common practices about network Anomaly Detection and Mitigation in 5G/B5G, CPS and IIOT, along with a presentation the attacks that commonly threaten these systems and networks. Additionally, a SotA and common practices for Cybersecurity incident detection and response modern frameworks is provided.

2.1 Cybersecurity attacks in IIoT and 5G

The following section presents a listing of known cybersecurity attacks against systems that utilize IIOT and 5G networks based on [21], [49], [50], [51], [52], [53], [54] . They are summarized in Table 1 Cybersecurity attacks faced by IIOT/5G networks. Cybersecurity attacks can be distinguished in Network attacks, Software attacks and Data Attacks and can be mapped against the seven layers of the OSI reference model [55]. The OSI model is presented in Appendix A. It should be noted that the specific scenarios and attacks that will be used to validate these modules in the project trials are still under development and will be reported in the next version of this deliverable, namely D5.10 “Anomaly detection and countermeasure selection tools.R2”.

Table 1 Cybersecurity attacks faced by IIOT/5G networks mapped to OSI layer

Type	Layer	Attacks
Network Attacks	Network Layer	Traffic Analysis, Wormhole, Sybil, RFID Spoofing and Unauthorized Access, LAN/WLAN spoofing, Routing Information, Man in the Middle/ Eavesdropping, Selective Forwarding, Replay, Sinkhole, Denial / Distributed Denial of services, Threats to Neighbour Discovery Protocol, Impersonation
	Session Layer	Denial / Distributed Denial of services

	Transport Layer	Desynchronization, SYN flood
Software Attacks	Application & Presentation Layer	Virus, Worms, Trojan Horses, Spyware, Ransomware, Cryptojacking
	All Layers	Misuse of audit tools
Data Attacks	Presentation Layer	Data Inconsistency
	Session Layer	Unauthorized Access (Remote to Local , User to Root)
	Network Layer	Data Inconsistency, Data Breach, Network virtualisation bypassing:
	Data Link Layer	Data Transit attack

Network attacks involve the manipulation of the functionalities of the network to gain access to sensitive or private or cause problems to the normal network operations.

- **Traffic Analysis:** Traffic analysis involves monitoring and analysing network traffic to gain insights into patterns, behaviours, and communication flow. Attackers may use this information to identify vulnerabilities or extract sensitive data.
- **Wormhole:** A wormhole is a network tunnel that connects two separate points, allowing data to be rapidly transmitted between them. In a security context, it can be exploited by attackers to bypass normal network security measures.
- **Sybil Attack:** In a Sybil attack, a single adversary controls multiple nodes on a network to manipulate communication and compromise its integrity. This attack is often seen in peer-to-peer networks.
- **Impersonation:** Impersonation involves pretending to be someone or something else to gain unauthorized access or deceive others in a network or system.
- **RFID Spoofing:** RFID spoofing is a special case of impersonation, which involves impersonating a legitimate RFID (Radio-Frequency Identification) tag to gain unauthorized access. This can be used to bypass security systems relying on RFID technology.
- **Unauthorized Access:** Unauthorized access refers to gaining entry to a system, network, or device without proper authorization. This can lead to data breaches, system manipulation, or other malicious activities.
- **LAN/WLAN Spoofing:** LAN/WLAN spoofing is a special case of impersonation, which involves creating a fake Local Area Network (LAN) or Wireless LAN (WLAN) to intercept and manipulate network traffic, potentially leading to unauthorized access or data compromise.
- **Routing Information:** Manipulating routing information involves altering the routing tables in a network to redirect traffic through unauthorized paths. This can lead to interception or disruption of data.
- **Man-in-the-Middle (MitM)/ Eavesdropping:** In a Man-in-the-Middle or Eavesdropping attack, an attacker intercepts and possibly alters the communication between two parties without their knowledge. This can lead to the unauthorized access of sensitive information.
- **Selective Forwarding:** In a selective forwarding attack, an attacker selectively forwards or drops specific messages in a communication network, leading to disruption or manipulation of the data flow.
- **Replay Attack:** A replay attack involves the interception and malicious retransmission of valid data, causing the system to repeat actions or responses as if they were original.
- **Sinkhole:** A sinkhole is a system or network component set up to redirect and capture malicious traffic. It is often used as a defence mechanism to mitigate the impact of certain types of attacks.
- **Denial of Service (DoS) / Distributed Denial of Service (DDoS):** DoS involves overwhelming a system, service, or network with traffic to disrupt or limit its functionality. DDoS involves multiple distributed sources coordinating these attacks.

- **Threats to Neighbour Discovery Protocol:** The Neighbour Discovery Protocol (NDP) is vulnerable to various attacks, including spoofing or manipulating Neighbour Advertisement and Neighbour Solicitation messages, leading to unauthorized access or network disruptions.
- **Desynchronization:** In desynchronization attacks, the intruder injects packets with fake sequence numbers of control flags that de-synchronize endpoints.
- **SYN Flood:** In a SYN flood attack, an attacker floods a server with a high volume of SYN requests, overwhelming its resources and causing it to become unavailable.

Software attacks, as the name implies are attacks that utilize software, many times self-propagating to exploit and introduce vulnerabilities into a system or interrupt its' normal operations.

- **Virus:** A virus is a type of malicious software that can replicate itself and spread to other computers by attaching to other programs. It can disrupt normal operations and may damage or delete files.
- **Worms:** Worms are like viruses but can independently replicate and spread across networks without attaching to other programs. They often exploit security vulnerabilities to move from one computer to another.
- **Trojan Horses:** Trojan Horses are deceptive software that appears legitimate but contains malicious code. Unlike viruses and worms, they don't replicate themselves. Instead, they trick users into installing them, often by disguising as useful or harmless programs.
- **Spyware:** Spyware is software designed to spy on a user's activities without their knowledge. It can capture sensitive information, such as login credentials or browsing habits, and send it to a third party.
- **Ransomware:** Ransomware is a type of malware that encrypts a user's files, making them inaccessible. The attacker then demands a ransom, usually in cryptocurrency, to provide the decryption key and restore access to the files.
- **Malware:** Malware is a general term for any malicious software designed to harm or exploit computers, networks, or users. It includes viruses, worms, Trojan Horses, spyware, ransomware, and other types of harmful software.
- **Cryptojacking:** Cryptojacking is a cyber-attack where an unauthorized entity exploits a person's or organization's computing resources to mine cryptocurrencies without their consent. This can achieve by injecting malicious code into websites, applications etc. causing infected devices to contribute computational power to cryptocurrency mining without their knowledge
- **Misuse of audit tools:** Mobile Network Operators (MNOs) employ audit tools to surveil network activities for optimization, security, and commercial purposes. However, these tools, containing data about the network and its users, pose a risk as malicious actors can exploit them for reconnaissance by leveraging insiders with privileged access within the MNO to extract sensitive information.

Data attacks are the types of attacks that affect data integrity or ownership in the system examined. These can be distinguished in three different types:

- **Data Inconsistency:** An attack against data integrity, e.g. the injection of fake data to a data stream. It leads to inconsistency in data transition and storage.

- **Unauthorized Access:** This attack involves using vulnerabilities to manipulate access control, to elevate the status of the attacker. Examples include a) elevation of the status of a user from remote to local or from simple to root user or b) disclosure of long-term keys for authentication and security controls conducted by an insider or hostile or untrustworthy personnel operating in the Core of a 5G Network. By gaining unauthorized access, malicious users can then try to gain data ownership or access sensitive data.

- **Data Breach:** Data breach or memory leakage refers the utilization of vulnerabilities that cause the disclosure of personal, sensitive, or confidential data in an unauthorized manner.
- **Network virtualization bypassing** This attack arises from poorly implemented or misconfigured slicing in a 5G network, risking data privacy breaches. In a shared network with various tenants, ensuring both the controlled entry/exit of legitimate traffic and preventing unauthorized slice access requires robust flow rule enforcement and protection against hostile actors exploiting hypervisor vulnerabilities at the core network level.

2.2 Cybersecurity incident detection & response

With cyber threats becoming more sophisticated and evolving, the creation and security of a modern Security Operations Centre (SOC) is crucial for organisations to protect against potential threats. The SOC [43] is a central hub of an organization's cybersecurity operations. Its primary objective is to detect and respond to security incidents and threats across the monitored infrastructure.

SOC is not necessarily a technological solution. It is a centralized function within an organization employing people, processes, and technology to continuously monitor and improve an organization's security posture while preventing, detecting, analysing, and responding to cybersecurity incidents.

As discussed in article [44], the future of SOCs aims to improve on the various points such as increased automation over detected attacks; the proper integration [45] and interoperability of different components (such as Security Information and Event Management (SIEM) [46], Security Orchestration Automation and response (SOAR) and Intrusion Detection System (IDS)/ Endpoint Detection and Response (EDR)); migration of SOCs to cloud-based environments (such as deploying SOCs in the cloud and including the ability to monitor modern networks, e.g. Kubernetes, deployed in cloud environments); improve collaboration between IT/OT/cybersecurity teams; focus SOCs on measuring metrics, e.g. the time between attack detection and response.

Regarding the integration and the interoperability of the different components, it is needed to mention that historically in the SOC one of the main problems is that they only use the SIEM. A Security Information and Events Management (SIEM) system was designed to collect, correlate, and store security events and generate appropriate security alerts for the operational needs. It is not sufficient [47] to meet the need to automate and orchestrate processes by connecting various tools through specific APIs to enable analysts to investigate and make decisions that increase the efficiency of incident response processes. This is where SOAR emerges [48], fulfilling these needs to orchestrate and automate cybersecurity events, adding the ability to provide an adequate response to them.

The modules developed in Zero-SWARM and presented in section 3 can fit in any modern SOC and will meet these milestones proposed by[44]. It should be noted that these components can be integrated directly into an industrial architecture such as zero-SWARM without the need to build a complete SOC to perform the cyber security detection and response functions. On the one hand, work will be done on the inclusion of different SOC components for the correct handling of cybersecurity events that may be found in CPSoS. However, taking into account that part of the Zero-SWARM architecture contemplates the cloud, a SOC will be developed that can be integrated in these environments and capable of analysing the traffic of networks peculiar to the cloud, such as Kubernetes networks. Finally, the SIEM and SOAR solutions presented in section 3.1, will focus on improving the metrics and representation of cybersecurity events so that they can be easily interpreted by operators with different profiles.

2.3 Anomaly Detection & Classification in 5G/b5G networks

The ongoing development of 5G networks is geared towards accommodating increased data capacity, a significant surge in the number of connected devices, more densely populated networks, improved connectivity even during high-speed mobility, all while delivering reduced latency and lower power

consumption compared to existing network generations. 5G upon fully realized deployment, will enable a wide array of vertical markets and industries to introduce a variety of new, diverse services tailored to different use cases along with new or not realized diverse threats against them.

Network anomaly detection in older generation networks is a well-studied subject and the approaches utilized can fall into three different categories [1]: a) Manual detection of the anomalies based on expert opinion, b) measuring some network traffic related value or values and based on predefined thresholds define normal and abnormal traffic and c) utilization of machine learning algorithms that are trained to recognize normal and abnormal traffic. The first two options are considered outdated [8] as they cannot keep up with the complexity and the scales of modern networks. However experimental studies [8] show that even ML based methods may not be able to keep up with the traffic density and throughput required by 5G / B5G networks.

Literature [8] suggests that deep learning is the ideal approach to address the problem of anomaly detection in 5G/B5G networks: Deep learning-based methods achieve SotA results in the task. Additionally, classic machine learning classification algorithms typically rely upon feature engineering methods to reduce the dimensionality of their input while Deep learning algorithms can automatically extract high-level features from large amounts of raw data, preventing overfitting based on regularization techniques [9]. There are numerous applications of DL algorithms to detect anomalies in 5G networks. In [1] the authors utilize a stack of self-attention based networks. In [8], the authors suggest a two-phase scheme: A Long Short-Term Memory Recurrent Network (LSTM) is used to detect the anomaly, while a DNN composed by a Deep Belief Networks (DBN) and Stacked AutoEncoders (SAE) is utilized to classify the anomaly into a specific category. The authors of [9] use a DBN to detect and classify the anomalies and showcase the efficiency of DNNs handling high dimensional data. In [10], the network flow data is treated as an image, inputted to a Residual U-Net Architecture, in an attempt to better model time dependent features without delays. In [11], a Variational Autoencoder (VAE) using Convolutional Neural Networks (CNN) is used to detect and classify the anomalies.

Finally, Deep packet Inspection (DPI) is an accepted approach to detect and prevent attacks such as worm propagation [12] in the operation of 5G networks. However, it has been pointed out [13], that while DPI offers a deep search for L2 (Data Link) up to L7 (Application Layer) flows it can be very resource consuming due to the large data volume expected in 5G/B5G networks. To remedy this, in [13] it is proposed to deploy DPI mechanisms in for specific cases that handle particular policy actions or events, e.g., after detecting a botnet in a specific RAN or to combine them into a multistage-stage detection process. Such a mechanism operating in multiple stages will be utilized in the Zero-Swarm project and is described in section 3.2.1. It is worth to mention that one of the major challenges in the industrial adaptation of DPI-based solutions is General Data Protection Regulation (GDPR) compliancy, fundamental rights (especially of Internet users), such as freedom of expression and privacy, as well as more economic concerns, such as competition and copyright [14] A pragmatic and effective solution besides the traditional technical challenges such as effectiveness and efficiency of operation, should consider solutions to ensure Data Protection Impact Assessment .

2.4 Anomaly Detection and Classification in IIoT networks and CPS

The potential of the so-called Industry 4.0, includes the promise to create manufacturing environments that are both real-time and secure, offering autonomy in their operations. The Industrial Internet of Things (IIoT) plays a pivotal role in turning this potential into reality by facilitating advanced wireless connectivity for seamless data collection and processing across interconnected industrial facilities and Cyber-physical Systems. However, implementing IIoT systems involves the integration of diverse technologies, resulting in the collection of data that may be incomplete, unstructured, redundant, or noisy. This situation gives rise to security vulnerabilities and challenges related to the quality of data within these systems.

To address these issues and ensure the integrity of the data, one effective approach is the deployment of anomaly detection systems. These systems provide specific insights to determine whether a device is malfunctioning, if a critical event is unfolding, or if there is a breach in the system's security. By employing early anomaly detection mechanisms, the IIoT system can avoid being influenced by anomalous data when making decisions. Additionally, early detection of anomalies in an industrial process is essential to implement decisions based on real-time information, thus reducing maintenance costs, minimizing machine downtime, increasing safety, and improving product quality [13].

A recent literature review of approximately 100 papers published after 2018 about IIOT anomaly detection [16], showed that DL methods make up ~56.5% of the proposed approach. Approximately, 22% of the remaining papers proposes statistics-based methods (e.g., Kalman Filters, Fourier Transformations, Markov Chains) and finally ~21% utilizes Machine Learning based methods (e.g., Decision Trees, SVM, DBSCAN). Concerning, the DL based methods [2], shows that Transformers are most common type of NN utilized, followed by Variational Autoencoders and LSTM networks.

The literature review conducted focused in AI based methods, since the Zero-SWARM project combines IIoT with 5G networks. Authors in [18] propose the use of Graph Deep NN for network anomaly detection and presents examples for three industrial use cases: smart transportation, smart energy, and smart factory. In [17] it is proposed to use a two-stage distributed approach that combines Autoencoder DNNs for traffic compression and the AdaBoost ML algorithm for the classification of the traffic to anomalous and benign. In [20] an approach is presented that utilizes Fully Connected DNN to detect anomalies in two Cyber-Physical Systems and showcases that generating synthetic adversarial data and retraining the DNN utilizing them, results to improved performance. In [21], the authors present a multi-stage, low latency module for a) the detection, b) the classification and c) the response against attacks against mission-critical Smart Factory Networks. The same paper shows that Multi-Layer Perceptron Models perform better from RNN, LSTM and CNN networks for anomaly detect, while CNN networks outperform the same networks for attack classification models. Finally, an Intrusion Response System (IRS) uses a predefined rule set where each type of attack is linked to a specific single countermeasure. The topic of proactive anomaly detection to secure the 5G enabled IIoT ecosystem is a hot topic of research and innovation in the ICT industry [22] . Ideally, there is a race to introduce proactive anomaly detection solutions that can automatically and adaptively introduce security measures for foreseen possible future attacks and enforce appropriate security measures to protect the IIoT network against them, even before they happen.

The mechanism proposed in section 3.2 combines the multi-stage approach of [17], [21] , using SotA algorithms and in contrast to [21] the IRS system will be able to select more than mitigation action for each attack type. Additionally, the training approach of [20], i.e., the utilization of synthetic training data creation will be also employed, however SotA Autoencoder models instead of the Fast Gradient Signed Method used in [20].

2.5 Countermeasure selection within cyber-systems

For the task of addressing threats within cyber-systems, including IIoT networks, two distinct alternatives for counter-measuring said threats exist. The first alternative focuses on addressing a specific type of attack, such as e.g., a wormhole attack that manipulates traffic flows through malicious network nodes to acquire access to data. This type of attack can be effectively countered as proposed in [23] by utilizing ML based techniques to recognize and block malicious network node.

The second alternative involves employing a system that strategically selects countermeasures based on the values of some Key Performance Indicators (KPIs) in response to the threats faced by the system. This approach allows for the simultaneous selection of countermeasures against attacks originating from multiple sources and involving various potential steps. Moreover, this approach is more holistic in the sense that it takes into account that an attack can be counter-measured by different mitigation measures, e.g. a DDOS attack can be mitigated by Rate Limiting, Packet Filtering or Host Isolation

[24]. Additionally, if this approach is automated, it can follow the so-called Autonomic cybersecurity paradigm.

A computing system can be called autonomic when it can automatically and dynamically configure and reconfigure itself under varying and even unpredictable conditions [25]. Focusing on cybersecurity, such a system should be [26]:

a) self-healing i.e., it should be able to discover and correct faults both in routine and unexpected without human interventions,
b) self-protecting i.e., it should automatically be able to detect and countermeasure attacks against its components,
c) self-configuring i.e., it should be able to configure/reconfigure itself in an automated manner to adjust to the available resources,
d) self-optimizing i.e., it should be able to monitor its' performance and continually seek to improve itself.

There are four distinct methods to utilize KPI based countermeasure selection. The first one involves measuring the values of one or more KPIs and presenting them to a human operator to manually select their desired actions.e.g., [27] presents such an approach to a SCADA system.

The second method involves automated mitigation using heuristic methods based on predefined thresholds for KPI values. The system, guided by predefined scenarios and values, selects actions from a list of predefined responses to counteract threats. However, this approach can result in intricate patterns of diverse cases and values, posing scalability challenges e.g., [28].

The third class includes approaches where the selection of mitigation actions is driven by optimizing a single KPI value or transforming the problem into a single objective (SO) problem, e.g., [29].

Finally, the fourth method comprises of approaches where the selection of mitigation actions is based on optimizing the values of multiple KPIs, which may be antagonistic but collectively offer a more comprehensive description of the action's impact on the system. An example of this approach is shown in [30] which uses evolutionary algorithms to solve the problem. The multi-objective optimization-based method is the one utilized by the module presented in section 3.2.2.

2.6 Anomaly Detection and mitigation in relevant standards

Anomaly Detection systems are more generalized forms of the so-called Intrusion Detection Systems (IDS). IEC TR 62443-3-1 [32] contains a section concerning Intrusion Detection Systems. There, such systems are categorized into two main types: Network IDS (NIDS) and Host IDS (HIDS). NIDS is most frequently deployed as a standalone device, such as when it is connected to a mirroring port on a network router or integrated within a router or firewall. NIDS is responsible for scrutinizing all network data to identify either known attack patterns or unusual and unexpected behaviour. On the other hand, HIDS is implemented as software on a host system and has the capability to inspect various sources, including logs, network traffic, and the file system, to detect signs of both completed and ongoing intrusions. A specialized variation of IDS can even take pre-emptive action against intrusion attempts, such as blocking network traffic associated with a detected intrusion effort. However, it is important to note that IDS systems have several limitations, primarily cantered around their cost, encompassing expenses related to deploying them across all subnetworks and hosts, ongoing monitoring costs, and dealing with false positives.

IDSs are considered among other practises (i.e. segregation of access, unique login accounts, password renewal) a crucial part of real-time monitoring of industrial systems according to numerous standards [32] - [39]. Some standards have more specific instructions. According to [40],[41] and [42] Passive monitoring solutions should be deployed both in the IT and OT environments to create an industrial

network traffic baseline and monitor anomalies and adherence to the baseline. The monitoring solution should also be deployed on the Access Layer to capture relevant internal traffic.

Additionally, [33],[36] and [37] propose that all events detected should be logged to enable analysis of events: To the extent possible, event logs should include user IDs, system activities, dates, times and details of key events (e.g. log-on and log-off times), use of privileges, etc. .

Finally, Deliverable D2.3 “Cyber-security implementation templates and methodological approach (Revised)” contains a table which describes the Security-by-design principles that are addressed by the modules described in this deliverable.

3 Zero-SWARM Anomaly detection & countermeasure selection tools

This section presents the modules developed for Zero-SWARM in T5.5, namely a module that handles Security Information and Event Management (SIEM) and Security Orchestration, Automation and Response (SOAR) procedures along with a module that monitors network traffic to detect network anomalies, including their type and then propose a set of actions to countermeasure these anomalies. This section provides a technical description of the modules: The interfaces used by the modules are described in D6.1 [5], while functional tests and related KPI for the modules were presented in D6.2 [6].

3.1 Cybersecurity incident detection & response

This subsection presents the cybersecurity incident detection and response system (see Figure 3) that will be developed and implemented in zero-SWARM.
This system is composed by two main components: the incident detection component and the incident response component. These two components oversee receiving information from a monitored endpoint. Also, there is a third component, the monitored endpoint.

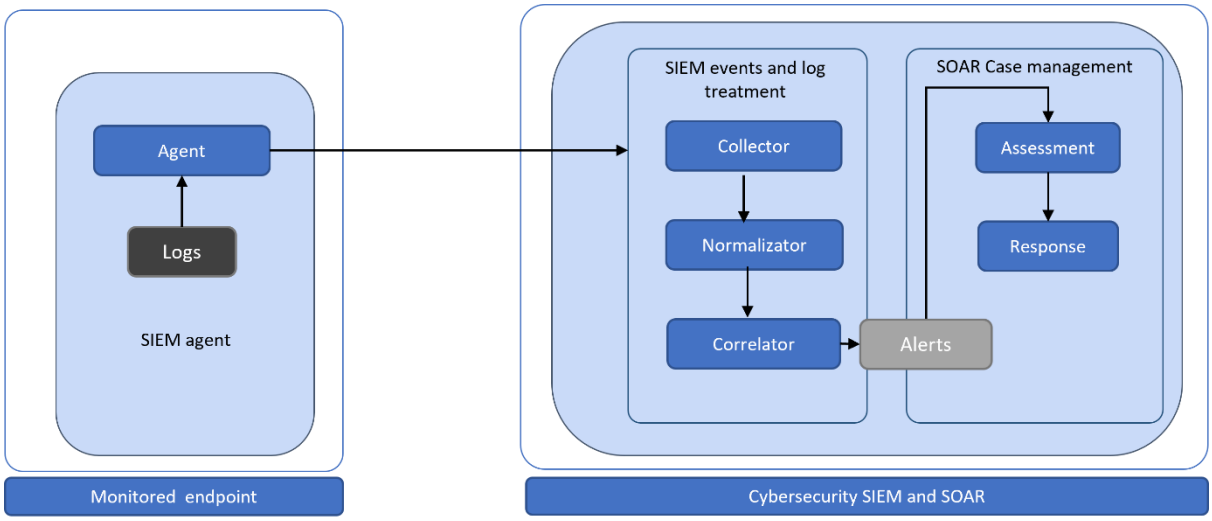


Figure 3 Cybersecurity incident detection and response in Zero-Swarm

The monitored endpoint is a network or a CPSoS, part of Zero-SWARM architecture or use case architecture that needs to be monitored to protect against cybersecurity threats. An agent will be installed in every monitored endpoint that oversees the collection of application logs, network traffic and analysing them for the detection of cybersecurity incidents.

3.1.1 Incident detection

In cybersecurity, detection is the ability to search for traces and identify possible attacks. The implementation of detection will be based on SIEM. SIEM technology supports threat detection, compliance

and security incident management through the collection and analysis (both historical and near real-time data) of security events, as well as a wide variety of other event and contextual data sources. The main capabilities are the collection and management of log events (such as data normalisation, enrichment with external sources, event correlation...) and the ability to analyse log events and other data across disparate sources, but also operational capabilities (such as incident management, dashboarding and reporting on detected events).

Combining security information management (SIM) and security event management (SEM), security information and event management (SIEM) provides near real-time monitoring and analysis of events and also provides tracking and logging of security data for compliance or audit purposes.

A security solution, SIEM helps organisations recognise potential security threats and vulnerabilities before they have a chance to impact business operations. SIEM detects anomalies in user behaviour and uses rules and decoders to automate many of the manual processes associated with threat detection and incident response. This has become a staple of modern Security Operations Centres (SOCs) for security and compliance management use cases.

As part of a SIEM component, a SIEM agent helps to standardise and provide different actions. A cybersecurity agent will perform the functions of an endpoint detection and response system, monitoring and collecting endpoint activity that could indicate a threat. The agent is installed on the network or host to be monitored (endpoint) by analysing network traffic passing through it and analysing log files with the intention of identifying anomalous behaviour that could be created by an attack on this endpoint. SIEMs have the capability of event ingestion by collecting raw data from the network and systems and event generation i.e. providing normalization and data aggregation capabilities.

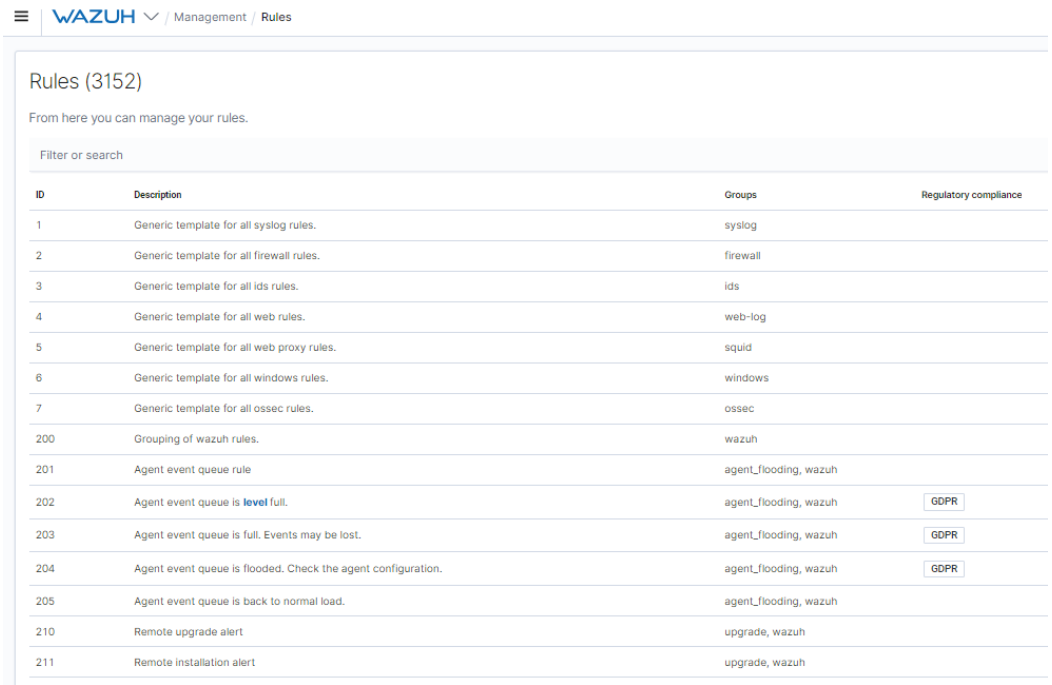


Figure 4 SIEM rules

To implement the SIEM functions, it has been decided to use Wazuh [56]. Wazuh is a free, open source security platform that unifies XDR and SIEM capabilities. It is currently the most advanced open SIEM platform and gives the possibility to integrate into different environments. This is why it has been decided to use this platform. The idea in this project is to use this component and expand its functionalities. Since this component will be positioned in the cloud (following the architecture presented in D2.3), the project will focus on packaging this software in Kubernetes[57] so that it can be easily deployed and integrated in cloud environments.

On the other hand, a direct integration with the SOAR component is not yet implemented. The project is working on the correct integration of the alerts generated by the SIEM in the SOAR. The idea of this integration is that the two components will be self-configuring once they have been installed without the need for extensive manual configuration work.

3.1.1.1 Event ingestion

Data ingestion to the SIEM is done from the SIEM agents. These agents monitor network traffic and applications at the Monitored Endpoint where they are installed. When these agents, by means of the rules they contain, detect a suspicious cybersecurity event, it is sent to the SIEM. Figure 4 shows an example of the rules contained in each SIEM agent. These rules are programmed in the SIEM and are distributed to each of the agents linked to it. These are the rules that the SIEM agent applies to the monitored traffic or logs. If any of these rules are met, the SIEM agent reports the log that has enforced the rule to the SIEM for analysis. In addition to this, the rules contain the configuration of the agents allowing them to be configured (and hot reconfigured if necessary) by the SIEM (central system).
The configuration for updating and managing the agents from the SIEM is done through the decoders. An example of these decoders can be seen in Figure 5. Decoders are the regular expressions that enable the SIEM agent to interpret incoming data traffic and application logs. In the same way as the rules, they allow the SIEM agents to be configured (also hot if necessary) so that they work correctly and capture the necessary data. In addition to using the default Wazuh rules and decoders, Zero-SWARM will proceed to develop new ones with the intention of adjusting them to the needs of the use cases.

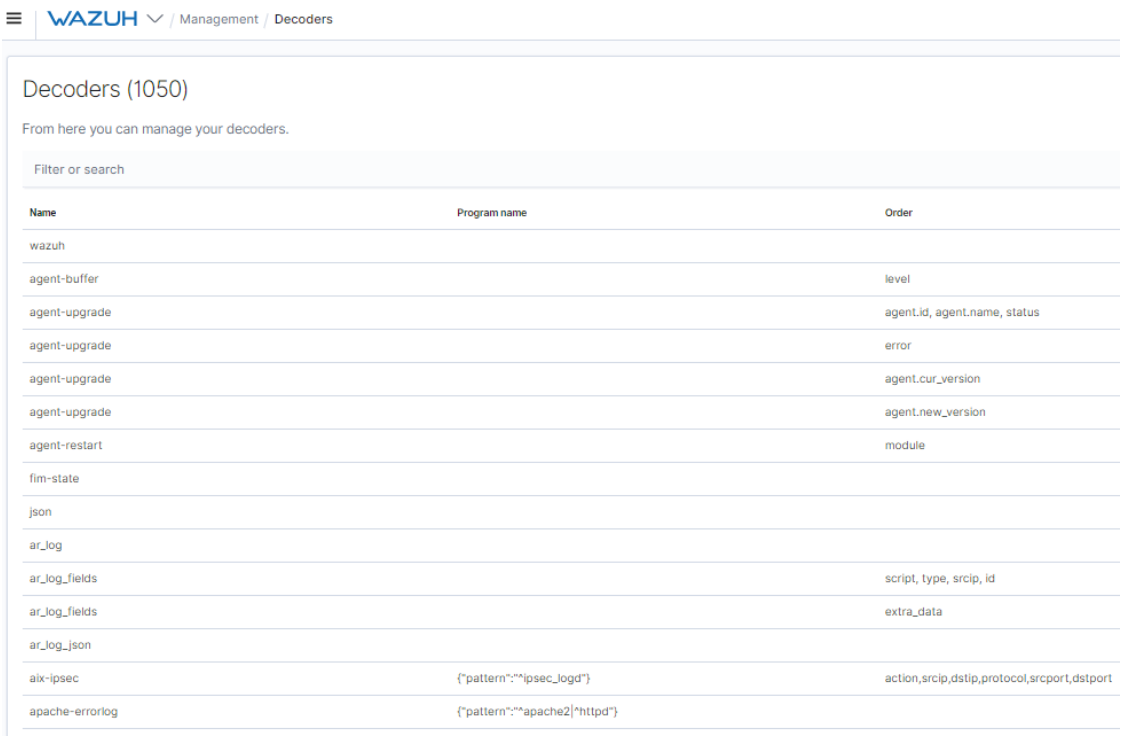
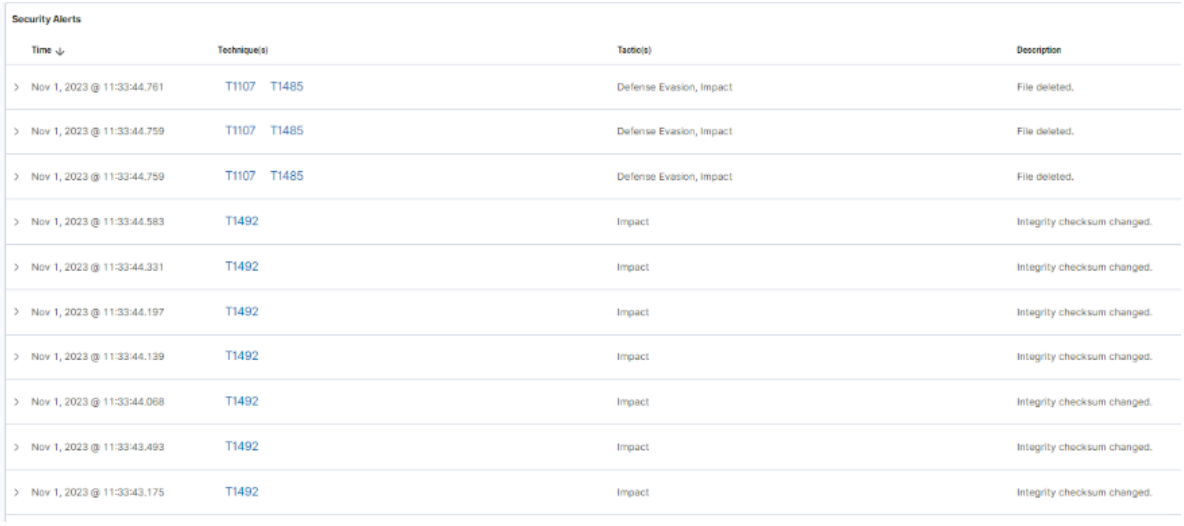


Figure 5 SIEM decoders

3.1.1.2 Event generation

Once the cybersecurity events generated by the SIEM agents have reached the SIEM, they must be normalised, aggregated into a database so that they can be analysed.

The aggregated events can be seen in Figure 6, where in addition to aggregating them, in this case they have been enriched with the MITRE [58] technique ID associated with the cybersecurity event detected. This helps the cybersecurity operator to save time in the implementation of the response to these events through the SOAR.



Security Alerts

Time	Technique(s)	Tactic(s)	Description
Nov 1, 2023 @ 11:33:44.761	T1107 T1485	Defense Evasion, Impact	File deleted.
Nov 1, 2023 @ 11:33:44.759	T1107 T1485	Defense Evasion, Impact	File deleted.
Nov 1, 2023 @ 11:33:44.759	T1107 T1485	Defense Evasion, Impact	File deleted.
Nov 1, 2023 @ 11:33:44.583	T1492	Impact	Integrity checksum changed.
Nov 1, 2023 @ 11:33:44.331	T1492	Impact	Integrity checksum changed.
Nov 1, 2023 @ 11:33:44.197	T1492	Impact	Integrity checksum changed.
Nov 1, 2023 @ 11:33:44.139	T1492	Impact	Integrity checksum changed.
Nov 1, 2023 @ 11:33:44.068	T1492	Impact	Integrity checksum changed.
Nov 1, 2023 @ 11:33:43.493	T1492	Impact	Integrity checksum changed.
Nov 1, 2023 @ 11:33:43.175	T1492	Impact	Integrity checksum changed.

Figure 6 SIEM aggregated events

In addition to representing the aggregated alerts, the SIEM visualises these alerts using a dashboard (as shown in Figure 7) that allows the cybersecurity operator to get an overview of the status of the monitored network.

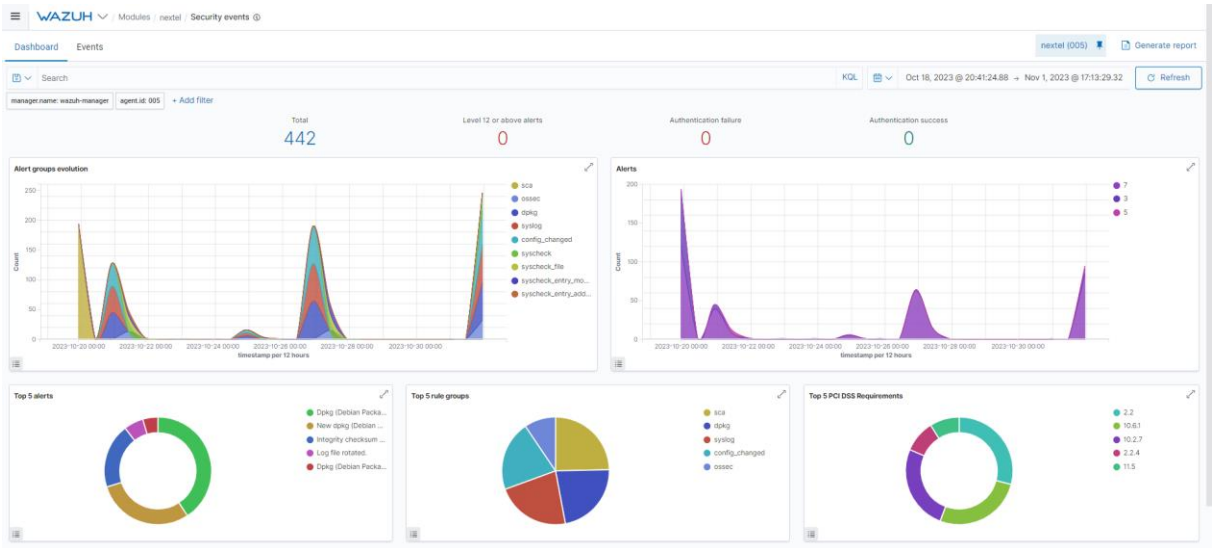


Figure 7 SIEM dashboard

3.1.2 Incident response

In cybersecurity, response is the ability to orchestrate the defensive actions when a possible attack is identified. The proposed implementation of response actions will be based on Security Orchestration, Automation and Response (SOAR) tools.

Security Orchestration Automation and Response (SOAR) is a suite of compatible software programs that enables an organisation to collect security threat data and respond to security incidents automatically or manually via an operator. The goal of using a SOAR platform is to improve the efficiency of security operations. Within this software stack are the orchestrator, automation, and response modules.

The Orchestrator Module connects and integrates internal and external tools through built-in or custom integrations and APIs. Connected systems can send alerts from endpoint protection products, firewalls, intrusion detection and intrusion prevention systems (IDS/IPS), SIEM platforms. In addition to alerts, external sources of threat information can be integrated.

With all the data collected, there is a greater chance of detecting threats, as well as more context and better collaboration. Inclusion of different sources requires a thorough analysis of alerts and linking them to threat intelligence sources. Security automation comes into play when security orchestration consolidates data to initiate response functions.

The Automation module, uses the data and alerts collected from security orchestration, to ingest, and analyse data and create repeated, automated processes to replace manual processes. These historically manually performed tasks such as vulnerability scanning, log analysis, ticket checking and auditing functions can be standardized and executed automatically using SOAR platforms. SOAR automation can make recommendations and automate future responses. Alternatively, automation can escalate threats if human intervention is necessary.

The Response module i.e. the SOAR provides analysts with a single view of the planning, management, monitoring, and reporting of actions taken once a threat is detected. It also includes post-incident response activities such as case management, reporting, and threat information sharing.

Depending on the type of alert, responses can be either automatic (solved by the execution of one or more responders, as a playbook once the alert has reached the SOAR) or manual (through the intervention of an operator to launch the necessary responses at any given moment).

3.1.2.1 Cybersecurity response implementation

This section describes how the security alerts are managed once they are ingested in the SOAR component. At this stage, SOAR can perform different tasks over the security alerts to improve the decision-making process regarding the resilience actions.

The SOAR that will be extended and implemented is based on TheHive [59] technology. TheHive collaborates an ecosystem of tools to provide this security alert management.

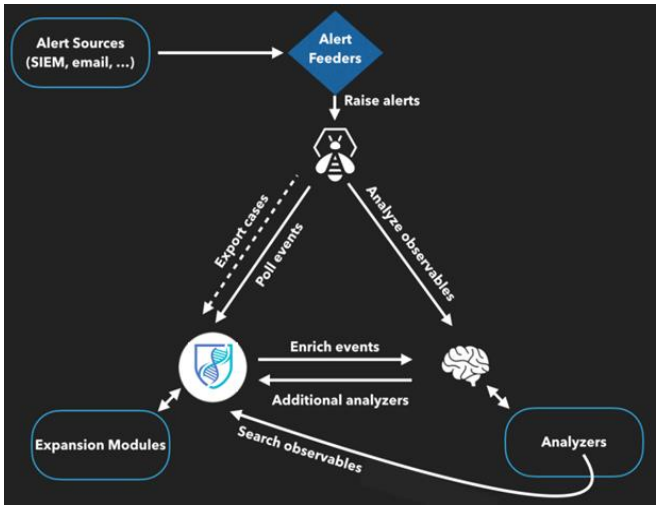


Figure 8 The Hive ecosystem[60]

As shown in the Figure 8, TheHive collaborates with other tools such as OpenCTI [61] for threat intelligence and Cortex [62]for observables management. This ecosystem allows different management features to the SOAR system to provide resilience action on the project, that are further described in the following subsections:

- Case escalation
- Alert enrichment
- Response/resilience action execution

It should be noted that these tools are not sufficient to manage and provide responses to cybersecurity incidents.

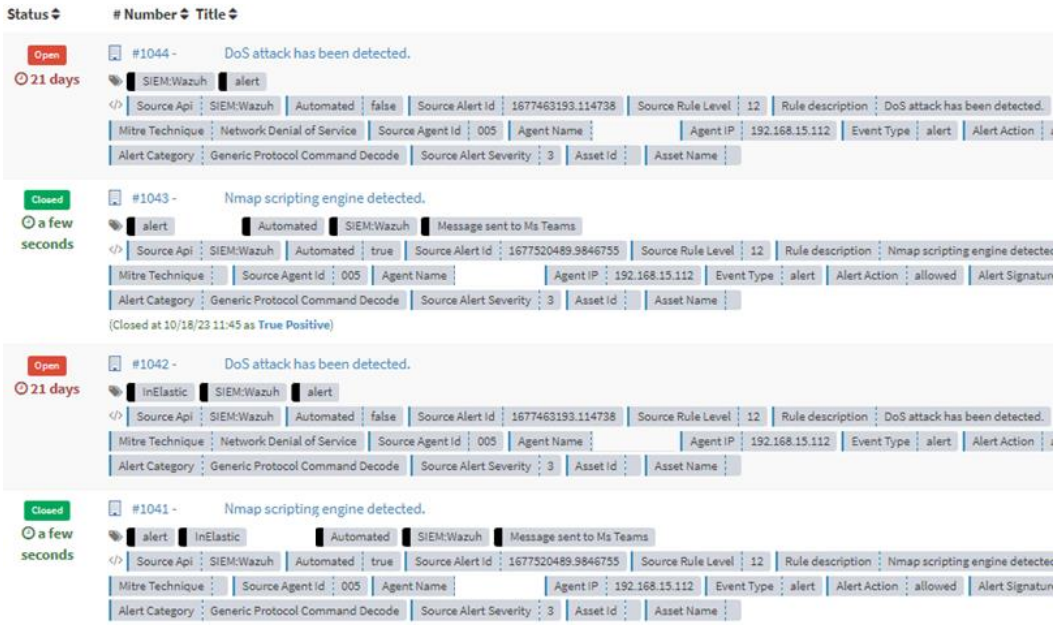


Figure 9 SOAR alerts

For this reason, the Zero-SWARM project will extend the functionalities of these components. In particular, new modules will be developed to connect SIEM and OpenCTI with TheHive: In addition to the connectors, most of the effort will be focused on the development of the responders that will allow mitigating the detected cybersecurity alerts.

3.1.2.1.1 Case escalation

Alerts can come to SOAR from different sources. In the case of zero-SWARM, alerts will come from the SIEM or SIEMs installed in different parts of the architecture. These alerts contain the information of the cybersecurity event detected in the monitored scenario The alerts that arrive and are normalised in the SOAR and are represented in the user interface that can be seen in Figure 9.

These alerts can be escalated to a case so that the cyber security operator can analyse them, relying on inputs provided by threat intelligence sources for example. As can be seen in the alerts, the SOAR automatically adds fields highlighting information that may be of interest, to facilitate the work of the cybersecurity operator and help him to resolve the alert in the best possible way.

3.1.2.1.2 Alert enrichment

As explained above, alerts are generated by the SIEM Agent which is installed on a monitored endpoint. These alerts are sent through the SIEM and if they are relevant, it creates an alert that is sent to the SOAR. In the SOAR this alert is represented and for cases in which this alert must be enriched, from here it is possible to access OpenCTI that will provide us with additional information to enrich the information of the alert.

In Figure 10, an example of a vulnerability that has been detected in a network component can be seen. This vulnerability has been injected into the SOAR as an alert and has been enriched with information obtained from OpenCTI.

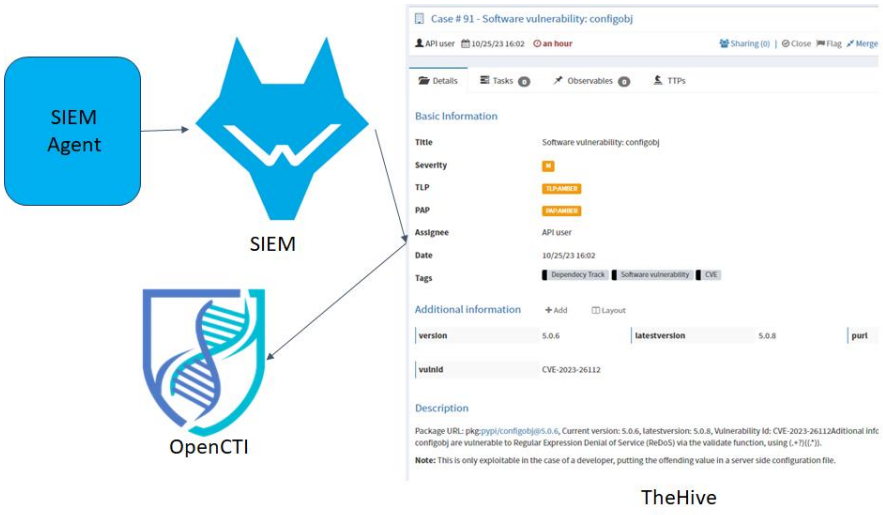


Figure 10 SOAR alert enrichment with OpenCTI

As mentioned above, the market does not currently have a functional connector to integrate OpenCTI information into The Hive. The connector between SOAR and OpenCTI illustrated in Figure 10 has been developed and implemented within the project. Figure 11 shows the detail of the alert enrichment information that has been applied through OpenCTI.

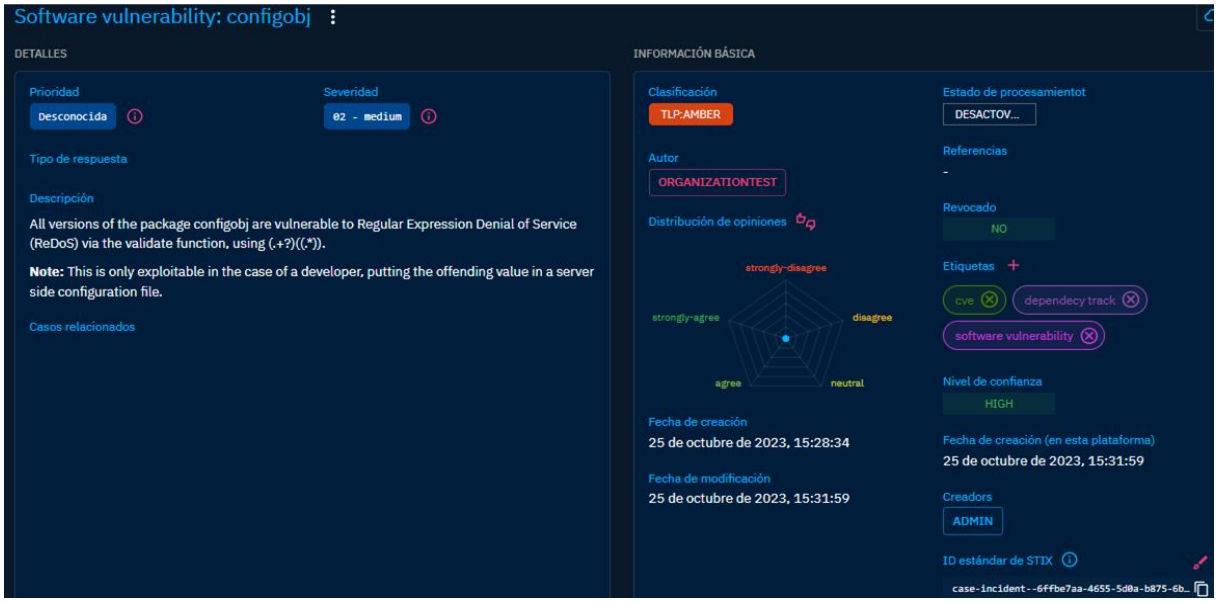


Figure 11 OpenCTI example detailed

In the next steps of the project, it will be analysed whether this information is sufficient or needs to include more CTI sources.

3.1.2.1.3 Response/resilience action execution

Once the alert has reached SOAR and has been normalised, integrated and displayed in the system, these alerts must be responded to in order to solve the associated cybersecurity problem. In order to implement these responses, the SOAR contains the Cortex component that is capable of launching responses in the form of cybersecurity countermeasures. In order to know to which object the Cortex response has to be launched, some observables are defined.

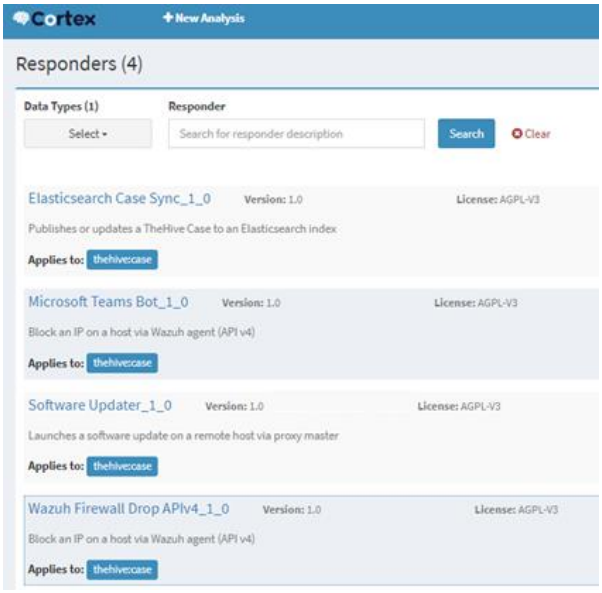


Figure 12 SOAR responders

These observables contain the necessary information, such as the IP address of the target machine, which allows Cortex to launch the response. In addition, Cortex contains the responders that are the components that are in charge of implementing the responses using the information from the observables. These observables and responders must be developed to meet the functionality required to implement cybersecurity responses. The interaction of The Hive, OpenCTI and Cortex can be seen in Figure 8.

At the moment, the resilience actions integrated in the Cortex implemented in Zero-SWARM are those shown in Figure 12, but during the project, and above all, after analysing the needs of the project's use cases, more responders will be designed and developed in order to implement them through the SOAR.

So far, the responders shown in Figure 12 have been developed and integrated into The Hive in the project. To meet the needs of the project's use cases, specific responders will be developed and implemented in the coming months. These responders will allow cybersecurity operators to better combat specific threats found in CPSoS.

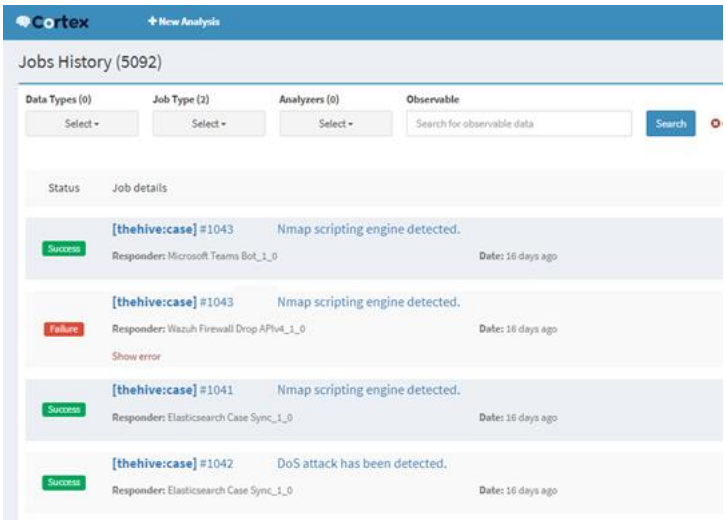


Figure 13 SOAR executed jobs history

Figure 13 shows an example of the history of the jobs carried out using Cortex. In these jobs we can see the response that has been launched. These responders can be triggered automatically (in case

the alert is known, and the remedy is known) or manually by a cybersecurity operator (who can use threat intelligence information to choose the best possible response to the alert).

The idea of developing specific responders focuses on minimising the time that elapses between the detection of a cyber-security incident and the time it is mitigated. For this reason, automatic responses to these will be implemented whenever possible, unless the alerts require the knowledge of a cyber-security operator. Implementing alerts automatically also helps to reduce the workload of cyber security operators by focusing their work on the more complex alerts that need special attention.

3.2 AI enabled Anomaly Detection & Mitigation Modules

The following section presents two modules that provide early cybersecurity related threat detection against the CPSoS along with mitigation of any detected attacks or malicious actions.

Anomaly detection can identify deviations from normal system behaviour, which could be indicative of a cyberattack or a system malfunction. Detecting these anomalies in their early stages allows for a swift response, minimizing potential damage or downtime.

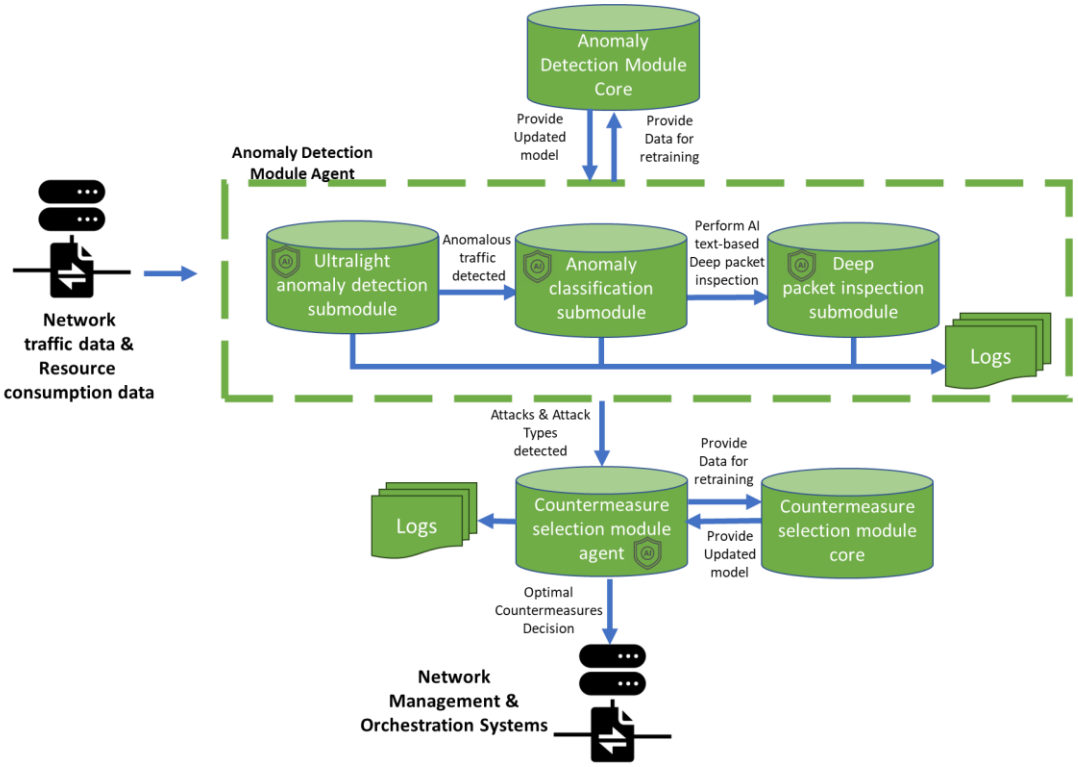


Figure 14 High level architecture and interactions of the AI enabled Anomaly Detection and Mitigation Module

IIoT networks face multiple, diverse attacks which can cause severe consequences ranging from monetary damages to physical harm if left unchecked. These attacks can be countered by multiple mitigation actions, creating the need for methods that help optimal mitigation action selection. Mechanisms are needed that automatically selects appropriate mitigation actions in an optimal way to countermeasure attacks faced by the System. Figure 14 provides a high-level overview of the architecture of the proposed module that handles these tasks . In the following subsections each of the modules is presented in detail.

3.2.1 Anomaly Detection Module

The following section contains a presentation of the architecture functionalities of the three submodules of the Anomaly Detection (AD) module: First the Ultralight Anomaly Detection (UAD) submodule which detects anomalies by monitoring network traffic. Then the Anomaly Classification (AC) submodule which discerns if the pattern of a detected anomaly corresponds to known attacks. Finally, the

Deep Packet Inspection (DPI) submodule tools allows the system operator to further gain knowledge about unknown anomalous traffic detected.

3.2.1.1 Ultralight Anomaly Detection submodule

Swift detection of the anomalies in the network traffic is a basis for securing the system as it allows for faster mitigation in the case of events caused by malicious actors. However, it should be noted that the purpose of this tool to detect anomalies, not their underlying cause: Anomalous traffic does not always indicate an attack, but it could also indicate malfunctions, hardware errors etc. Network traffic anomaly detection has been extensively researched and multiple approaches have been suggested including statistical and ML methods. The available literature suggests that AI extends better to large-scale intrusion data with higher dimension compared to 'traditional' ML methods [31].

The submodule receives Raw traffic data as input, captured in the form on. *. pcapng files. A high level overview of the functionality and I/O of the submodule is shown in Figure 15. Initially the following traffic related features will be taken into consideration, as proposed by [63] :

- Basic Flow Features: Destination Port, Protocol, Flow Duration, Total Forward Packets, Total Back-ward Packets, Flow Pkts/s
- Inter-Arrival Time (IAT) Statistical Metadata: Flow IAT Mean, Flow IAT Standard Deviation, Flow IAT Maximum, Flow IAT Minimum, Flow IAT Total, Forward IAT Mean, Forward IAT Standard Deviation, Forward IAT Max, Forward IAT Min, Backward IAT Total, Backward IAT Mean, Backward IAT Standard Deviation, Backward IAT Max, Backward IAT Min

The features that will be used in the final version of the tool will be decided by experimentation.

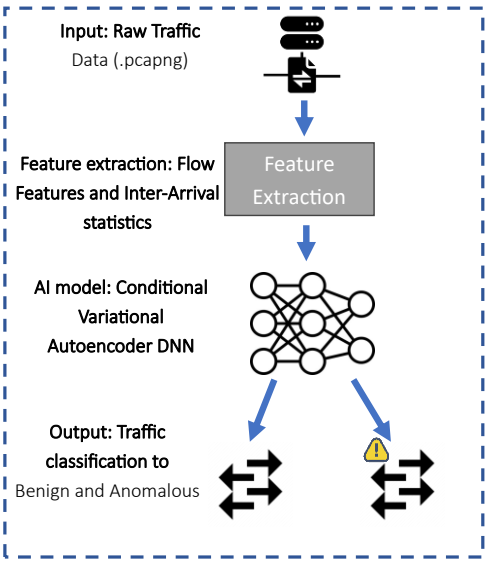


Figure 15 High level overview of the Ultralight Anomaly Detection module

We propose the use of a SotA Conditional Variational Autoencoder (CVAE) DNN variant, the "Log-Cosh" CVAE [64] for the task of anomaly detection. CVAE models the distribution of observed data via latent variables in an unsupervised manner. It can efficiently handle imbalanced classes and high dimensional data, which is essential in the anomaly detection problem since benign traffic outweighs anomalous traffic. The architecture of the DNN is shown in Figure 16.

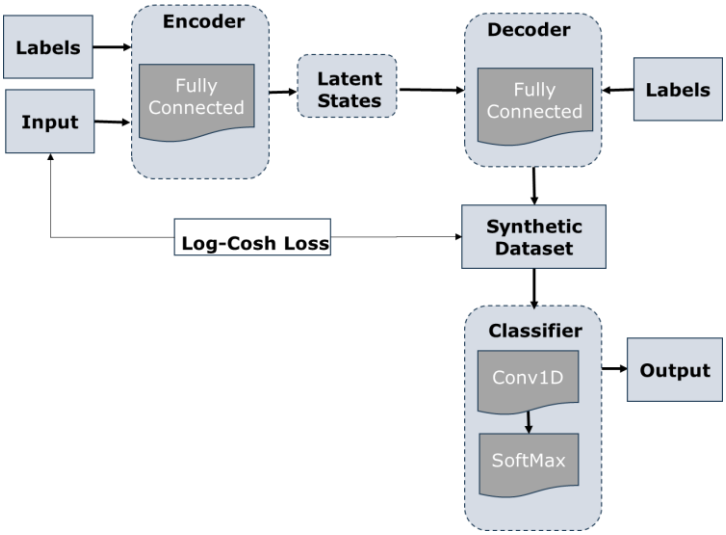


Figure 16 CVAE DNN model architecture

Let $X \in R$ be the input data containing Basic flow data and Inter-Arrival Time (IAT) Statistical metadata described earlier above and $C \in \{Benign, Anomalous\}$ the class labels associated with with it. Let $Q_\varphi((Z|X,C))$ and $P_\Theta((X|Z,C))$ be the Encoder and Decoder networks shown in Figure 16 and ϕ,θ be learnable parameters.

The purpose of the encoder is to use X, C to create a latent variable $Z \sim Q_\varphi((Z|X,C))$. Then in the decoding phase, the latent variable and $c \in C$ are utilized as an input to generate new samples for label c, $\hat{X} \sim P_\Theta((X|Z,C))$.

The Log-Cosh CVAE utilizes the following objective function:

$$L_{log-cosh}(X,\hat{X}) = \frac{1}{a}\sum_i^l log(cosh(a(X_i - \hat{X}_i))),$$

where $a \in R$ is a hyperparameter, $X_{i,} \in X$, $\hat{X}_i \in \hat{X}$ are the ith elements of X and $\hat{X}$ respectively.

Finally, the comprehensive loss function for the model is

$$L(\varphi,\theta;X,C,a) = L_{log-cosh}(X,\hat{X}) - D_{KL}\left[Q_\varphi((Z|X,C))|P_\Theta((X|Z,C))\right]$$

where D_{KL} is the Kullbach-Leibler divergence, which allows us to measure how much Q and P differ.

Collecting benign traffic data for a network is a trivial process. By generating $\hat{X}$ for anomalous traffic data we can create a labelled dataset S that is balanced for all $c \in C$, created by merging, $\hat{X}, X, C$. This is utilized to train a lightweight 1D Convolutional Neural network (CNN) that performs binary classification. This classifier is then used to discern between anomalous and benign traffic. This the same approach proposed in [64], which to our knowledge has not been tested in IIoT and 5G network traffic.

To evaluate the model, in D5.10 we will utilize typical binary classification performance measures such as Accuracy, Precision, ROC etc. along with measurements of the time required to classify a traffic sample. Appendix B presents a comprehensive list of the performance measures that will be used to access the success of the classification.

3.2.1.2 Anomaly Classification submodule

After an anomaly is detected by the UAD, the next step is to try and discern a) the type of anomaly and b) in the case of a cyber-attack/cyber-threat, its' type. Correct recognition concerning the types of attacks faced by the system is essential to select the appropriate countermeasures to mitigate them.

Moreover, this tool helps to monitor cases of false positives: Benign traffic identified as anomalous might indicate that the UAD submodule needs retraining.

The submodule receives the segments of traffic recognized as anomalous by the UAD and uses the same features as it (Basic Flow Features and IAT metadata) to try and classify the anomalous traffic to three classes:

1. Attack when the pattern of the traffic corresponds to a known attack type,
2. Benign when the pattern of the traffic corresponds to non-malicious/normal traffic i.e., in the case the UAD produced a false positive.
3. Uknown, when the pattern of the traffic does not correspond to any of the previous cases, which warrants more inspection by the system operator.

Figure 17 presents a high-level overview of the functionality and I/O of the submodule.

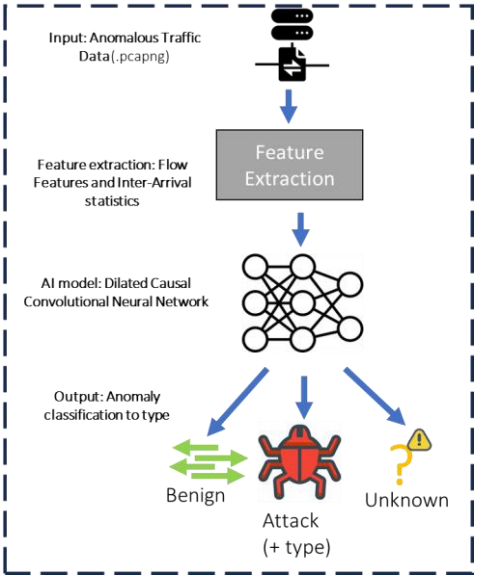


Figure 17 High level overview of the Anomaly Classification submodule

We propose the use of the Dilated Causal Convolutional NN (DCCNN) [69] to handle this task. These types of networks are also known as Wavenets or Temporal Convolutional Networks. Convolutional Neural Networks (CNNs) segment the input data using so called filters, which allows them to learn specific patterns. Contrary to other types of DNN e.g., LSTM (Long-Short Term Memory) DNNS, CNN are by design not fully connected, meaning that the not all nodes of the network relate to one another.

This means that less calculations are required thus CNN are usually less computationally expensive to fully connected DNNs of similar size. Simple CNN have been shown to effectively model multi-dimension patterns. They can also capture the high temporal correlation of traffic data. Additionally, CNN models scale better compared to RNN [70]. A special case is the Dilated Convolutional Neural Network: In this variant of the CNN, the filters are applied by skipping certain elements in the input, allow the receptive field of the network to grow exponentially [70] . This property allows them to model even sparce data along with both long-term and short sequence relationships present.

The architecture of the DCCNN is presented in Figure 18 along with an example of standard and dilated convolution. It should be noted that the number of dilation layers and the dilation rate we will be using is still under investigation and will be determined experimentally.

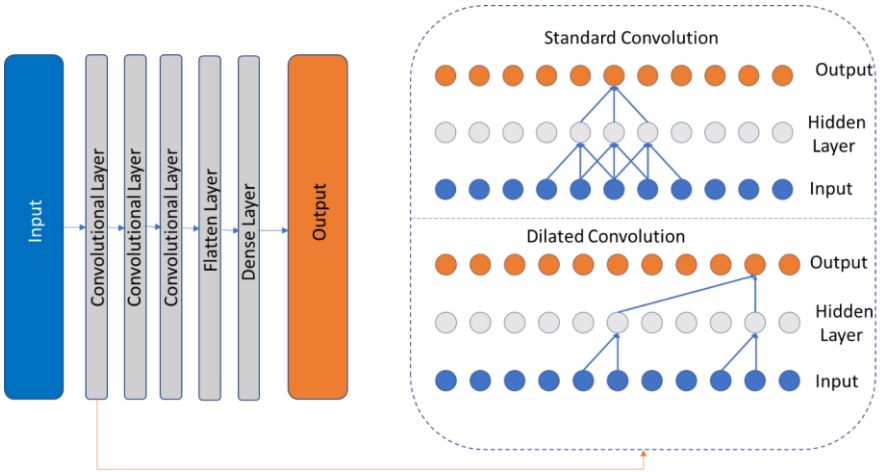


Figure 18 Architecture of the Convolutional Neural Network along with an example of standard convolution and a dilated convolution with a dilation rate = 4 for the grey layer and 1 for the blue layer

Let $X = \{X_{t-(k-1)}, X_{t-(k-2)}, X_{t-1}, X_{t-(k-1)}\}$ be the input data, i.e. a timeseries of consequent observation of anomalous traffic data with $k \in N$ being the kernel size, $t \in N$ being the time window size i.e. the number of observations in the time-series and O_t the output obtained using the values from X. The dilated causal convolution operations over the network layers is described by the following equation [71]:

$$x_l^t = g\left(\sum_{k=0}^{K-1} w_l^k \, x_{(l-1)}^{(t-(k*d))} + b_l\right)$$

Where x_l^t is the output of the neuron at position t in the lth layer; K is the width of the convolutional kernel; w_l^k stands for the weight of position k; d is the dilation factor of the convolution; and b_l is a term representing bias.

3.2.1.3 Deep Packet Inspection submodule

In the constantly evolving field of cybersecurity, system owners and malicious actors constantly try to outperform one each other. System owners seek to boost the defence of their systems e.g., utilizing smarter systems, training employees to cybersecurity procedures, and designing more secure architectures while malicious actors try to discover new vulnerabilities and attacks. One such new attack type, are the so called Advanced Persistent Threats (APT) [73]: APTs are highly sophisticated and targeted cyberattacks typically conducted by well-resourced adversaries, such as organized hacking groups commonly known to target IIoT systems [74] . APTs involve a stealthy and prolonged intrusion into a specific organization's network or systems, with the goal of gaining unauthorized access, remaining undetected for an extended period, and often exfiltrating valuable data or causing other harm. These threats require advanced tools and techniques, adaptability, and a strategic focus on their victims, making them challenging to detect and defend against.

In this section we propose a tool that will allow the operator to use traffic traces and the content of the packets exchanged, even if they are encrypted, to discover APTs. An indication of such threats

would be anomalous traffic segments that were classified by the Anomaly Classification submodule as Uknown. Figure 19 High level overview of the Deep packet Inspection submodule.

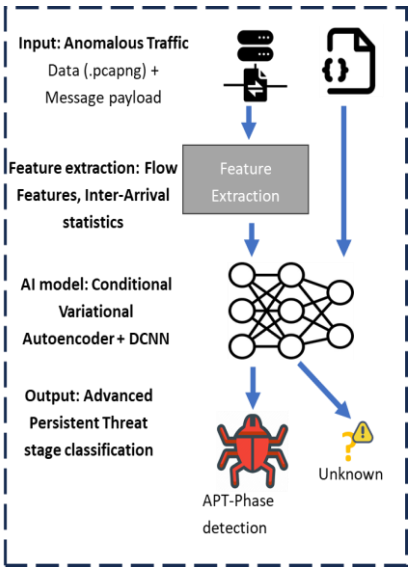


Figure 19 High level overview of the Deep packet Inspection submodule

APT Attacks can be split to four distinctive stages:

1. Reconnaissance, where mapping of network topology, available services and used software versions to detect possible entry points.
2. Foothold establishment: entry points are exploited to setup a command-and-control channel.
3. Lateral movement: use a compromised system to deeper penetrate the system and access systems that are not directly connected to a public network.
4. Data exfiltration: Stealing non-public, possible high value data.

The tool will operate on daily time granularity as proposed in [80]: We propose the use a combination of Log-Cosh CVAE and DCNN to expand the results of [80], which utilizes a simple CNN for the classifier. Figure 20 presents the proposed architecture of the DPI submodule.

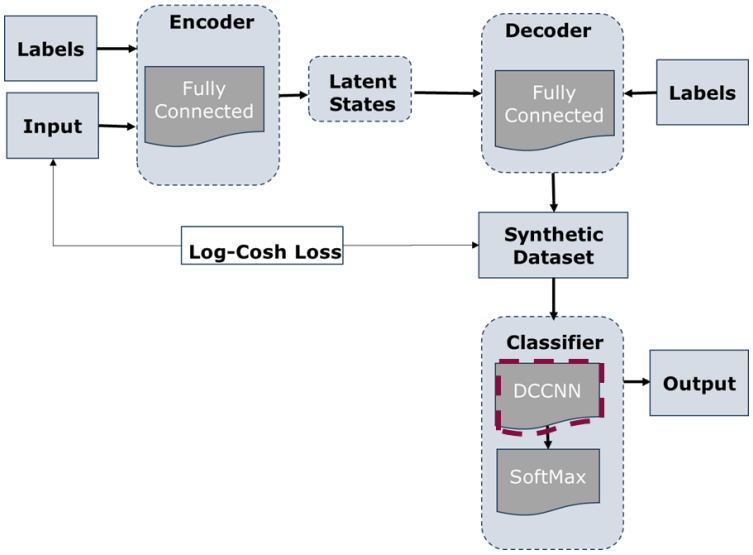


Figure 20 CVAE combined with DCCNN model architecture.

3.2.1.4 *Datasets used to train the Anomaly Detection Module*

To train and evaluate the anomaly detection submodules multiple datasets will be utilized. These datasets are publicly available and cover more than 40 different attacks in different networks (5G, IIoT, IoT) and layers. These are presented in Table 2.

Table 2 Publicly available datasets that will be utilized to train and evaluate the Anomaly detection submodules.

Dataset (network type)	Attacks
5G-NIDD (5G) [72]	8 attack types categorized as DOS, Port Scans
5GAD (5G Core) [75]	6 attack types categorized as Reconnaissance Attacks, Network Reconfiguration Attacks, DOS
M2M using OPC-UA (IIOT) [76]	3 attack types i.e., DOS, Man-in-the-middle and Spoofing/impersonation
DS2OS (IoT application Layer) [77]	7 attack types categorized as Network Scans, Malicious operations, DoS, Data probing, Reconfiguration
CIC IoT Dataset 2023 (IoT) [78]	31 attack types categorized as Brute Force, DDoS, Spoofing, DoS, Recon, Mirai, Web-Based
InSDN (IoT/SDN) [79]	17 attack types categorized as DOS/DDOS, Password guess, Webapp attack, Botnet attack, SDN specific attacks
APT-2020 (APT) [80]	4 stages (Reconnaissance, Establish Foothold, Lateral Movement Data Exfiltration) and 4 attacks (Sql Injection, Portscan, Xss, Bruteforce)

3.2.2 Countermeasure Selection Module

Timely detection of cyber-attacks is crucial to enable swift mitigation, as delayed recognition of a security breach can lead to significant repercussions. Establishing robust and secure IoT platforms and networks offers substantial benefits for both industry stakeholders and end-users. To effectively combat the growing array of attacks aimed at the ever-changing landscape of IIoT networks and devices, it is imperative to create innovative, intelligent solutions capable of addressing the multifaceted nature of these attacks and either thwarting or mitigating them. Artificial Intelligence (AI)-based approaches serve as a solid foundation for such tools.

In the following sections, we present an approach using a multi-objective (MO) method, based on a Deep Neural architecture called Pointer Networks [65], for optimizing the selection of countermeasures of a network used in a CPSoS under attack. Countermeasure selection using MO techniques is well studied in 'classic' computer networks, however it is a new topic in the context of IIoT. Previous worked is expanded: In previous related work [66] , the normalized normal constrained (NNC) method was used as a general framework to generate final set of solution. While NCC is intuitive in its' application, it is known to suffer for two major drawbacks: First, the method is heavily dependent on the choice of the Utopia Points and second is that in some cases it can produce non-pareto solutions that then require filtering which adds latency to the system. Both drawbacks can be overcome by utilizing decomposition methods [67] instead of NCC. I/O data for the AI mechanism.

This subsection presents the input and output data used the proposed AI algorithm: These all also presented schematically in Figure 21. For each device type that is anticipated to be part of the IIoT

network, a variety of mitigation measures exists that can be described as a set of predefined rules. These rules are established by the CPSoS operator, drawing from their knowledge of past or known attacks, and include the following information: a) a unique Device Identifier, b) the Device Type, c) details regarding vulnerabilities, d) appropriate mitigation actions for addressing these vulnerabilities, and e) values for the variables necessary for calculating Key Performance Indicators (KPIs) that will be utilized for optimization purposes. It's important to note that a single device may have multiple rules, each outlining distinct actions to counteract an attack targeting a particular vulnerability. We assume that some mechanism is utilized for monitoring device types and their identities, with unidentified devices being automatically categorized as 'unprofiled'.

If one or more attacks are detected against a network of N interconnected devices, the rule table associated with these attacks will consist of M rows. The input data for the proposed method is presented in the form of an ordered matrix with dimensions MxN, encompassing multiple rules for various devices. Each row in this matrix corresponds one-to-one with the original table and contains N elements that describe the attributes of the respective device.

Let $M = \{p1, p2, \ldots, p|M|\}$, be the ordered vector corresponding to each row of the input matrix. Then, the output of the Pointer Networks is a vector P of size equal to M,

$P = \{c1, c2, \ldots, c|M|\}$, where

$$c_m = \begin{cases} 1, \text{if pointer } p_m \text{ is selected as part of the solution,} \\ 0, \text{if not.} \end{cases}$$

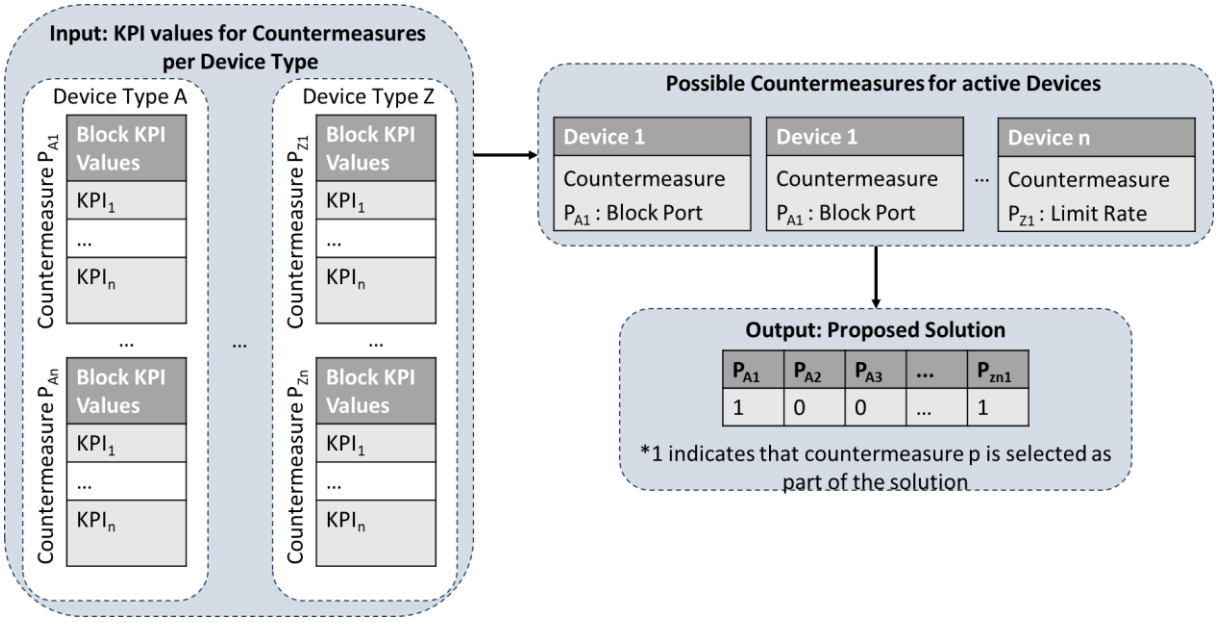


Figure 21 Input and output for the proposed AI Algorithm

The proposed method considers two constrains a) one mandatory that enforces that only a single mitigation action should be applied to each device and b) possible constraints imposed by the operator concerning the values of the KPIs used.

3.2.2.1 Methodology used for the Unsupervised Countermeasure Selection Engine

In the proposed method, a deep neural network architecture called Pointer Networks is utilized to solve the problem of selecting the set of optimal countermeasures.

First, the constrained multi-objective optimization (MO) problem is defined: Let x be n items we must choose from, xi = 1 if an item is chosen else xi = 0, and: let μl be the lth KPI to be optimized. Then, the following minimization is the target:

$$\min_x(\mu_1(x) \dots \mu_l(x)), l \geq 2$$

Let $g_j(x)$ and $h_k(x)$, be the r and s inequality and equality constrains of the problem:

$$g(x) \leq 0, (1 \leq j \leq r) \quad (1)$$
$$h_k(x) = 0, (1 \leq k \leq s) \quad (2)$$
$$x_i \in 0,1 \quad (3)$$

Equations (1), (2), (3) are inequality, equality, and side constraints of the problem respectively. To proceed, n subproblems are be defined and solved, one for each objective function:

$$\min_x \mu_1(x), (1 \geq i \geq l)$$

Each sub-problem is subject to equations (1), (2), (3).

3.2.2.2 Solution of the MO problem

Pointer Networks are utilized to solve equation (9) : Let P= {p1, … , pn} be a sequence of n vectors transformed via a Linear embedding to be used as input, corresponding to x = {x1,...,xn} and Y = {y1,...,ym} be the output sequence associated to P. Pointer networks are based on the Sequence to Sequence Model [68] with a modified attention mechanism. Bi-directional Long Short-Term Memory (LSTM) DNN are used to encode and decode the data:Let E= {e1, … , en} and D= {d1, … , dn} be encoder and the decoder hidden states of the LSTMs. Let f, g be the transformation functions made by the LSTM layers, c be a context vector resulting by an attention mechanism q(e1,…,ej). Then, the conditional probability calculation can be written as (i, ∈ (1,…,n)):

$$p(y_i|y_1, \dots, y_{i-1}, P) = g(y_{i-1}, d_i, c) \quad (10)$$

$$di = h(d_{i-1}, y_{i-1}, c_i) \quad (11)$$

$$c = q(e_1, .., e_j) \quad (12)$$
$$e_j = f(P_j, e_{j-1}) \quad (13)$$

In Pointer networks, the Encoder and Decoder Layer are connected by the attention mechanism. The context vector c are calculated by the encoder hidden states and the attention weights values aj , i,j ∈ (1,...,n):

$$c_i = \sum_{j=1}^{n} a_j^i e_j, (14)$$

where

$$a_i^j = softmax(u_i^j) = \frac{\exp(u_j^i)}{\sum_{k=1}^{n} exp(u_k^i)} \quad (15)$$

and

$$u_i^j = v^T \tanh(W_1 e_j + W_2 d_i) \quad (16)$$

with W1, W2 and v parameters that are learned by the network.

Finally,

$$p\big((y_i|y_1,\dots,y_{i-1},P;\theta)\big) = softmax\big(u^i\big) \quad (17)$$

provides the conditional probability to choose the pointer yi at a given iteration of the algorithm. The SoftMax normalizes the Attention Layer vector values u_i^j to be an output distribution over the dictionary of inputs. After a pointer is chosen, the constrains set by equations (6), (7), (8) are checked and if a violation is found x and $p\big((y_i|y_1,\dots,y_{i-1},P;\theta)\big)$ are assigned to zero, else xi = 1. Figure 22 shows the architecture of the Pointer Deep Neural Network. The outputs of solving the sub-problems defined in equation 9, can be utilized as input to a decomposition method that can be used to find the final set of optimal solutions. An approximation of the final set of optimal solutions can be decomposed into a number of scalar objective optimization sub-problems. For the Countermeasure Selection Module, the penalty Boundary Intersection proposed in [67] will be used.

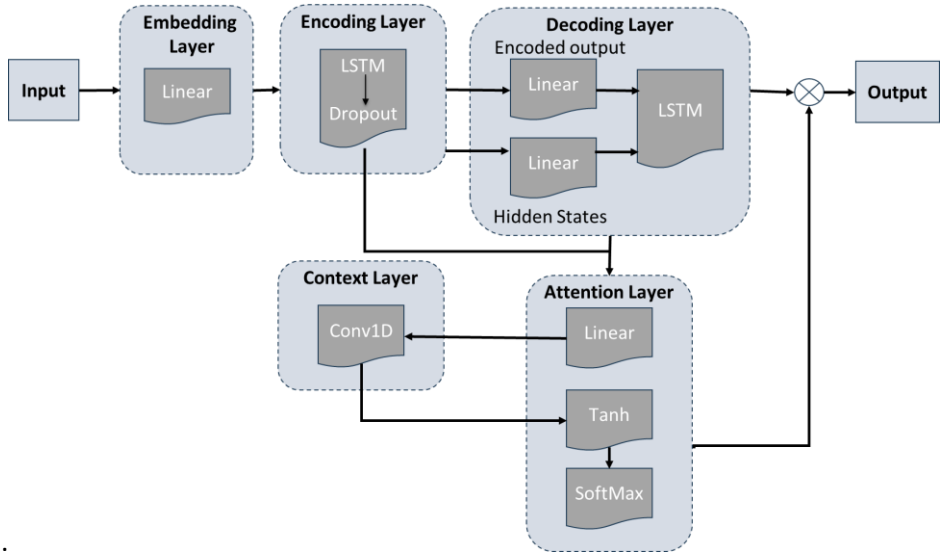


Figure 22 Architecture of the Pointer Deep Neural Network

4 Integration to IEC61499 Simulation environment

Section 3 focused on the presentation of the modules that are developed in the project while section 4 focuses on presenting details concerning integration and validation efforts. Initially, we present the plans concerning the integration of the modules introduced in section 3 to the IEC 61499 platform created in T5.1 and detailed in D5.1-Distributed automation and information management. The purpose of this integration is to evaluate and verify the functionalities of the modules developed in T5.5 in a realistic manner before deploying and testing them in the Zero-SWARM trials. Additionally, a preliminary laboratory setup, used for initial evaluation purposes of the SIEM/SOAR module detailed in section 3.1, is presented. Finally, we briefly present tools that are commonly used for real time monitoring network traffic, and we justify the choice of one of them as the input source for the Anomaly Detection module detailed in section 3.2.

4.1 Integration of the IEC 61499 platform in the Anomaly Detection

The following section presents a IEC61499 simulation platform and the plans and architecture for the integration of the modules developed in T5.5 to this platform. The platform simulates parts of the Schneider Electrics EcoStruxure Automation Expert. This is a new category of industrial automation systems with IEC61499 at its core. It enables automation applications to be built using asset-centric, portable, proven-in-use software components, independent of the underlying hardware infrastructure. It also allows the user to distribute applications to any system hardware architecture of choice —

highly distributed, centralized, or both — with minimal to no additional programming effort. Finally, it supports established software best practices to simplify the creation of automation applications that interoperate with IT systems.

EcoStruxure Automation Expert is a cohesive system consisting of a suite of integrated hardware and software solutions, built around the EcoStruxure Automation Expert engineering, monitoring, and management environment Distributed Programmable Automation Controller (dPAC) platforms with a common, flexible, scalable runtime across:

- Schneider Electric hardware:
  - ATV dPAC for Altivar
  - Modicon M251d/TM3 I/O
  - Modicon M580d/X80 I/O
- Innovative new software-based controllers:
  - Soft dPAC for Linux™
  - Soft dPAC for Windows™
- EcoStruxure Automation Expert - HMI, a fully integrated, object-orientated industrial visualization solution
- EcoStruxure Automation Expert - Archive, a centralized solution for the historization of process data, alarms, and trends
- Schneider Electric Libraries, a comprehensive set of hardware-independent libraries, ranging from basic functions to segment solutions.

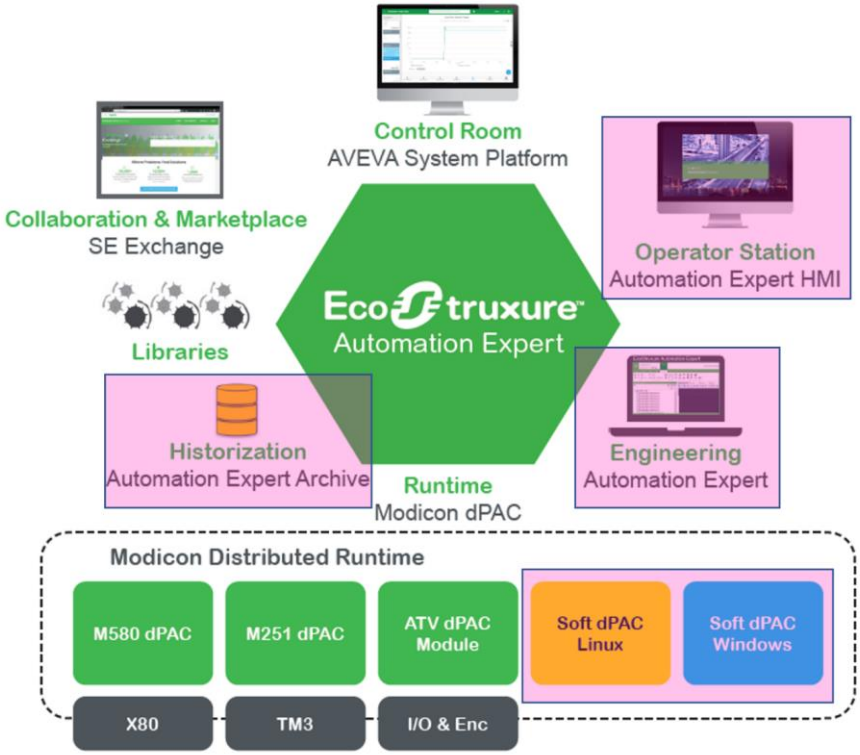


Figure 23 Schneider Electric EcoStruxure Automation Expert

The simulated testbed integrates the Operator Station HMI, the automation Expert, and its' Archive along with softwarized dPACs for both Windows and Linux OS. The entire system is shown in Figure 23 (above) and the simulated testbed components are marked pink. The component integration in the simulated testbed is shown in Figure 24 (below).

For the simulated testbed, an IEC61499 test application was developed that simulates the operation of an industrial environment, with the goal to utilize all IEC61499 software modules, as depicted in the Figure 24, and showcases the network architecture in relation to the 3-tier architecture of the zero-SWARM use cases outlined in D5.1. For the test application details see deliverable D5.4. We aim to simulate the asset centric approach, to distribute the application on two tiers (Edge Device layer and Edge Gateway layer) and we simulate the 3-tier architecture, by encompassing various communication loads and protocols with IT clients on the cloud tier. This test bed will be utilized specifically during the anomaly detection scenario where the employed protocols, which are based on Ethernet/TCP or UDP standards, are analysed for more specific anomaly information. A brief outline of the operations that will occur is provided in the next paragraph.

In Figure 24 the secured connections between the primary IEC 61499 test application on the "**edge device**" layer and the MQTT Broker and OPC UA Client on the "**edge gateway**" layer and the cloud layer are exchanging data during anomaly detection tests. The same is happening between the Secondary test application on the edge gateway layer with the same MQTT and OPC UA counterparts. Between the primary and secondary test application a distributed system is simulated with IEC61499 cross communication UDP messages, which is not secured. Simulated values are stored in an EAE archive over a secured connection and the EAE HMI displays the simulated values over a secured connection. The EAE Engineering workstations deploys the projects of the primary and secondary test application over a secured connection to the edge devices and edge gateway layer. All network connections to the different EAE modules are watched by the anomaly detection modules.

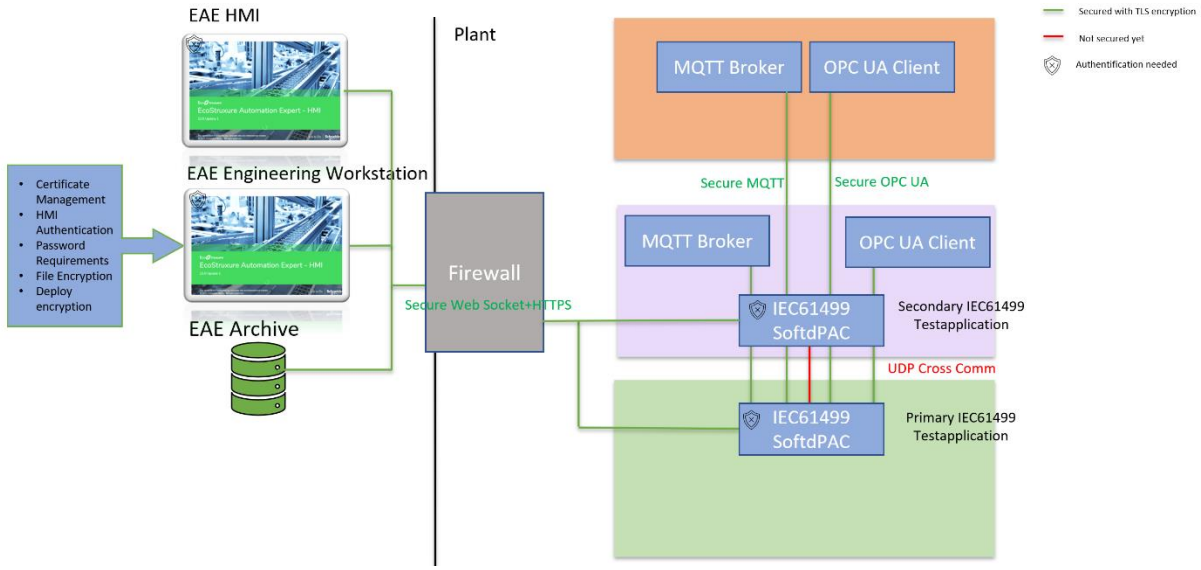


Figure 24 IEC61499 platform and network architecture for simulation environment

A high level overview of the connection of the systems and modules developed in T5.1, T5.4 and T.5.5, presented in D5.1, D5.4 and this deliverable are shown in Figure 25 & Figure 26: Attacks and anomalous events will be introduced utilizing the various data simulations developed in D5.4 (OPC UA, MQTT, Modbus). Moreover, we plan to integrate the anomaly detection module to the SIEM system and the Countermeasure selection system to the SOAR.

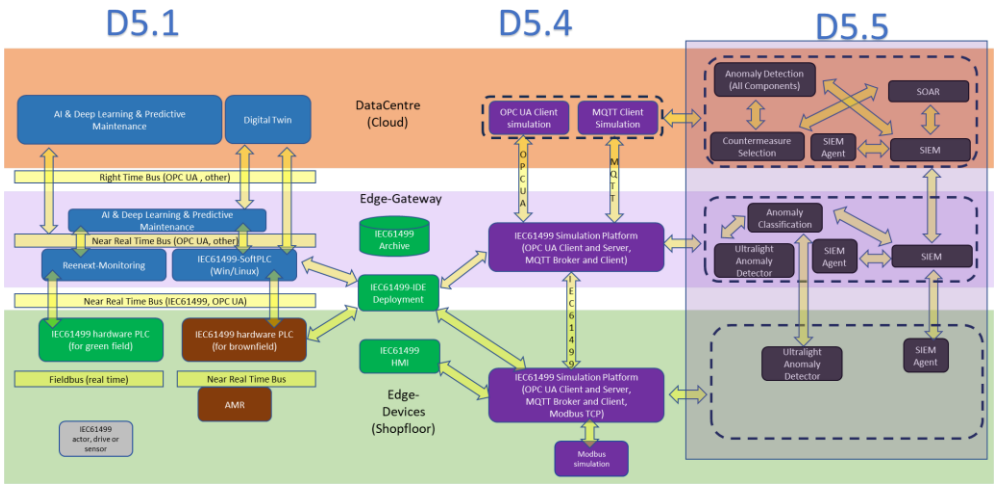


Figure 25 Interconnection between the environment and modules developed in T5.1, T5.4 and T5.5 and their respective deliverables.

The IEC61499 runtime platform, which is located on the edge device layer and the edge gateway layer, where the test applications are running, measures specific KPIs on the IEC61499 platform itself. These KPIs could be an indicator for the anomalies, and we plan to perform a correlation analysis with the anomaly detection measurements.

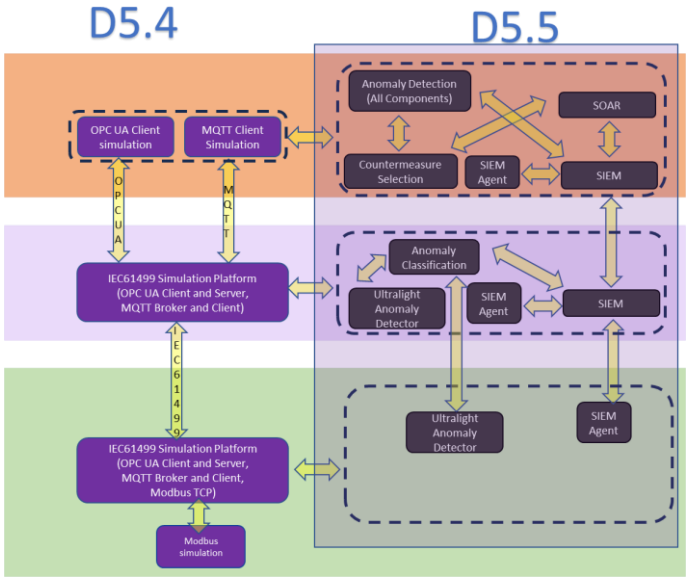


Figure 26 Zoom-in to the interconnection between the environments and modules developed in T5.4 and T5.5 and their respective deliverables.

A brief description of the KPI measured by the IEC61499 runtime platform is provided in the following list:

- **CPU load [in %]**: When two PLCs communicate, the CPU load of each PLC may be impacted. CPU load refers to the percentage of processing power utilized by the PLC's central processing unit. The communication-related tasks of IEC 61499 RT, such as data transmission, protocol handling, and message processing, consume CPU resources and other tasks of the operating system included in this load. Effect on CPU Load: PLC communication can increase the CPU load due to additional processing required for handling communication tasks of IEC 61499 RT.

The complexity of the communication protocol, the volume of data exchanged, and the frequency of communication all contribute to the CPU load. Higher communication demands can result in increased CPU utilization and potentially impact the execution of control logic. In the application, the CPU load is not impacted much due to less process and memory consumption by the processing unit.

- **Memory used [in Kbytes]**: The KPI for the statistics about memory consumption by the IEC61499 process. Monitoring memory usage helps ensure that the PLC has adequate memory resources to handle communication requirements efficiently. Memory usage shown in the chart mentioned as “PhysMemoryUsed” is showing the total usage by the runtime and the application consumption together. The active usage of memory by IEC 61499 RT is monitored by applying the command “/proc/meminfo” while the communication process takes place between two IEC 61499 RT-based PLCs.
- **Event-Latency [in seconds]**: Event latency measures the time it takes for a PLC to detect an event or trigger and respond accordingly within the IEC 61499 Runtime. It is a critical KPI for assessing the responsiveness and timeliness of the IEC 61499 control system.
- **Response-Time [in milliseconds]**: Response time measures the time it takes for an IEC61499 PLC to respond to a communication request of another IEC 61499 PLC. It includes the transfer time of the IEC61499 message from primary test application to the secondary test application and back. Monitoring and optimizing response time is essential for achieving timely and efficient communication between PLCs and detect additional influences of other IEC61499 processes, like the service communication from the shop floor to the IT (edge to cloud).

4.2 Cybersecurity incident detection and response preliminary laboratory setup

In the Figure 22 the architecture of a preliminary laboratory setup of a Cybersecurity incident detection and response component is presented.

In this first deployment in the S21Sec lab, we decided to monitor a host with a Linux operating system, where we installed an intrusion detection system (IDS) that is able to capture and analyse the traffic on the host's network interface. In addition, a specific process has been created to monitor the logs of the installed applications to detect malfunctions. These two components report the logs/intrusions to the SIEM agent, which is responsible for sending them to the SIEM component.

he SIEM component normalises this data, correlates it with historical data and displays it on a dashboard for the operator to analyse. In addition to this, the SIEM can detect cybersecurity threats in this data, and for each of these, it creates an alert that is sent to the SOAR. The SOAR is responsible for orchestrating this alert input, enriching it with external data (such as threat intelligence sources) and managing the alerts. To solve the detected cyber security problem, the SOAR can launch responses (in cyber security countermeasures mode) via the SIEM agent automatically or manually, through interaction with the cyber security operator.

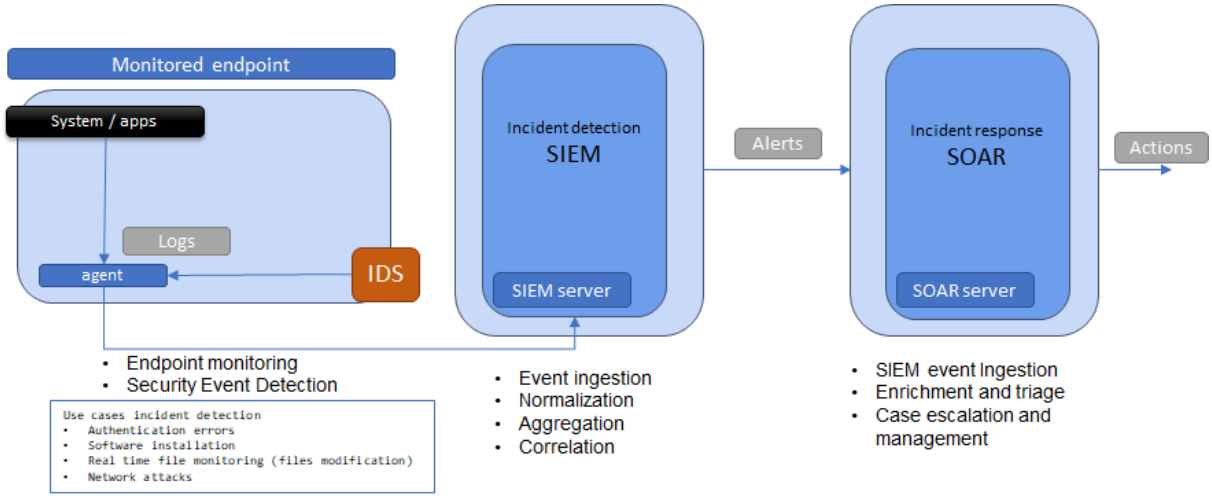


Figure 27 Cybersecurity incident detection and response

4.3 Network traffic monitoring and capture tools

This subsection briefly presents and compares the two tools most used to capture and analyse network traffic in real-time, TCPdump and Wireshark along with a command-line based variant of Wireshark, called Tshark. This data will be used as an input to the Anomaly detection module described in section 3.2.1.

4.3.1 TCPdump

TCPdump is a command-line packet analyser that is part of the Linux family of operating systems. Operating as a packet sniffer, it intercepts and records data packets traversing a network, facilitating a detailed examination of network communications. It offers real-time capture of packets, allowing users to inspect live digital interactions within a network. TCPdump demonstrates proficiency in handling diverse communication protocols utilized by networked devices during data exchange. This enables precise protocol analysis and aids in deciphering the nature of transmitted information.

TCPdump offers filtering options, allowing users to selectively capture packets based on specific criteria such as source or destination IP addresses, protocols, or port numbers. TCPdump is a command-line interface-based tool, offering a lightweight yet versatile solution for packet capture since it allows scripting. In summary, TCPdump serves as a proficient and versatile packet analyzer, allowing real-time data capture and protocol analysis.

4.3.2 Wireshark & Tshark

Wireshark is a network protocol analyser that serves as a tool for the comprehensive examination of network communications. Functioning as a packet sniffer, it captures and dissects data packets in real-time, providing an intricate view of the dynamics within a network. It exhibits proficiency in understanding diverse communication protocols, allowing for the precise interpretation of digital conversations. Key functionalities include advanced filtering mechanisms, enabling selective isolation of packets based on specified criteria such as IP addresses or protocols. The tool employs a visual representation with color-coded packets, facilitating efficient identification of different protocols. Additionally, Wireshark allows for in-depth packet content inspection, elucidating the nature of transmitted data. Its utility extends beyond real-time analysis, incorporating post-mortem statistical reporting for retrospective examinations.

Tshark, is the command-line variant of Wireshark. Functioning as a packet sniffer, Tshark shares core functionalities with its graphical counterpart, Wireshark, while offering the advantage of command-line efficiency. The primary functionality of Tshark lies in its real-time data capture capabilities. Like Wireshark, it intercepts and records data packets traversing a network, enabling a granular examina-

tion of digital communications. Analogous to Wireshark, Tshark incorporates advanced filtering mechanisms, permitting users to selectively capture packets based on specified criteria such as IP addresses, protocols, or keywords. Tshark's command-line interface provides a streamlined and efficient approach to packet analysis, particularly advantageous in scenarios where a graphical user interface is impractical or resource-intensive. This characteristic renders it suitable for deployment in a range of environments, from server configurations and containers to remote or headless systems.

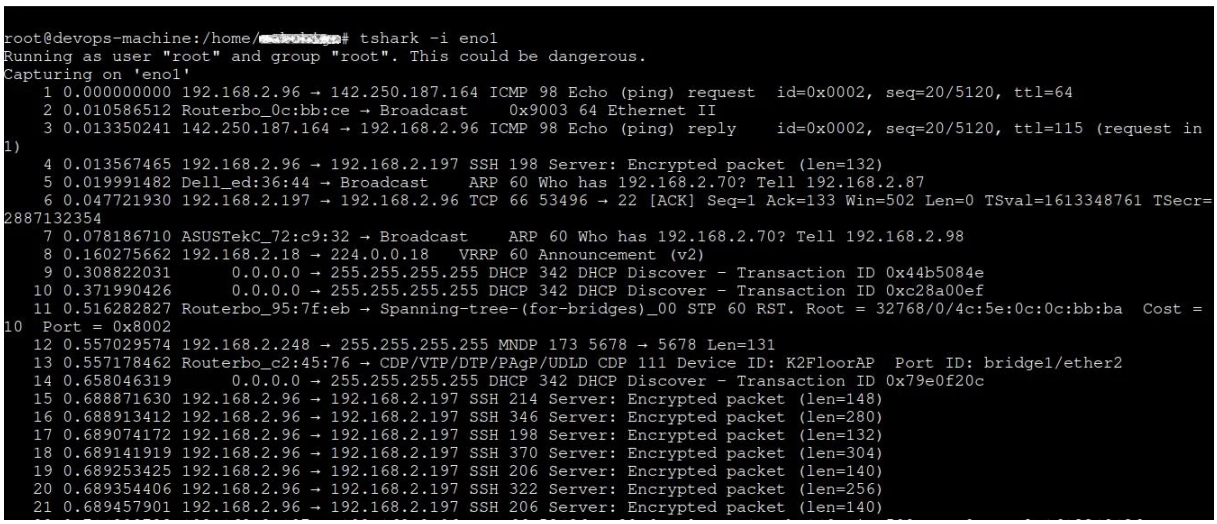


Figure 28 Example of utilizing Tshark and an onboard Network Interface Card to capture traffic

4.3.3 Comparison of tools and justification for selecting Tshark

Traffic monitoring and capture is essential for the anomaly detection module presented in section 3.2.1. More specifically, its Ultralight Anomaly detection submodule promises to detect network-based anomalies in milliseconds. To achieve this, an efficient method to capture traffic data in real time and then serve it to the module is required. Moreover, for the Anomaly classification submodule (Figure 14), we need a tool that can capture segments of suspicious/anomalous traffic data. Both Wireshark, TShark and TCPdump allow storing traffic data in the pcapng format. This stands for PCAP new generation and is the to-go file format for saving traffic traces[81]: it builds on and remedies multiple shortcomings of the pcap file format such as the inability to store packets with different link layer types.

However, the module will be available via Docker containers. This excludes the use of interface-based tools such as Wireshark Tshark has been selected due to the following features [82][83]:

a) it can support significantly more protocols than TCPdump,
b) it offers more efficient data filtering functionalities and
c) it deals better with encrypted data which will be useful for the Deep Packet Inspection submodule.

For these reasons the Tshark tool will be utilized for network traffic monitoring and storing by the Anomaly Detection module.

5 Conclusions

The traditional segregation between Industry and Operational Technology (OT) environments and Information Technology (IT) is evolving as industrial systems increasingly integrate IT technologies for enhanced efficiency. This convergence, however, exposes industrial infrastructures to security vulnerabilities due to the interconnection of components lacking robust security measures. This heightened connectivity poses a significant risk, exposing production and manufacturing processes to potential

threats from the IT and Internet realms. Anomaly detection and mitigation mechanisms are crucial in safeguarding the integrity and security of industrial systems.

Given the rising sophistication of cyber threats and the increased connectivity of industrial systems, robust defences are imperative. Modern anomaly detection methods employ advanced algorithms to continuously monitor network traffic, system behaviour, and data patterns, identifying any unusual activities that may indicate a cyber-intrusion. Upon detecting anomalies, immediate mitigation measures, such as isolating compromised systems or updating security protocols, can be applied to prevent potential damage or data breaches. In the current era where industrial systems are prime targets for cyber-attacks, anomaly detection and mitigation mechanisms play a vital role as frontline defences, ensuring the resilience and reliability of essential systems.

The work presented in this document dealt with the high-level technical details of anomaly detection and countermeasure selection mechanisms developed in Zero-SWARM. This deliverable aims to provide the technical background and description of these modules. It should be noted that the interfaces used by the modules are already described in D6.1" Integration, validation & specification of the trial demonstration", while various functional tests and related KPI for the modules were presented in D6.2 "Trial demonstration & evaluation results".

The inclusion of the SIEM and SOAR components in the zero-SWARM project will allow us to protect the key components developed in the project from cybersecurity attacks. To this end, these components will be monitored by means of SIEM, carrying out an exhaustive analysis of network traffic and the logs generated by the applications to identify any anomalous operation that could be linked to a cyber-attack. The SOAR will be responsible for orchestrating and launching a rapid response with the intention of minimising the damage that the suspected attack may cause to the monitored components.

These components will be integrated with two AI enabled Anomaly Detection and Mitigation Modules that will utilize state of the art deep learning algorithms to provide a fast and efficient mechanism that monitors traffic in real time to detect cyberattacks attacks and proposes optimal countermeasures for them.

The integration of the modules presented in this deliverable with the IEC61499 simulation platform developed in T5.1 will allow partners to further research on the subject of anomaly detection in industrial communication protocols, such as Modbus, OPC-UA, MQTT and IEC61499 communications in an environment, closer to a real industrial production line.

The first versions of the modules will be available in the end of M21, while the integration with the IEC61499 simulation platform with take place in M22, paving the road for demonstrating the modules in the trials of the project. Final development and validation of the developed modules will be reported in the next version of this deliverable, namely D5.10 "Anomaly detection and countermeasure selection tools.R2": that deliverable will contain the final technical description and details of the developed mechanism, the details on the integration between the AI enabled Anomaly Detection and Mitigation mechanism and the SIEM/SOAR mechanism, along with experimental results that validate the effectiveness of these mechanisms.

References

[1] Zero-Swarm project, “D2.2 - Eco-design architecture, specifications & benchmarking.R2”, 2023

[2] Zero-Swarm project, “D2.3 - Cybersecurity implementation templates & methodological approach_RV1”, 2023

[3] Zero-Swarm project, “D5.1 - Distributed automation & information management final”, 2023

[4] Zero-Swarm project, “D5.4 - Penetration & hypothesis testing diagnostic plugins”, 2023

[5] Zero-Swarm project, “D6.1 - Integration, validation & specification of the trial demonstrations”, 2023

[6] Zero-Swarm project, “D6.2 - Trial demonstration & evaluation results”, 2023

[7] L. Lei, L. Kou, X. Zhan, J. Zhang, and Y. Ren, “An Anomaly Detection Algorithm Based on Ensemble Learning for 5G Environment,” *Sensors*, vol. 22, no. 19, p. 7436, Sep. 2022

[8] L. Fernández Maimó, Á. L. Perales Gómez, F. J. García Clemente, M. Gil Pérez, and G. Martínez Pérez, "A Self-Adaptive Deep Learning-Based System for Anomaly Detection in 5G Networks," in *IEEE Access*, vol. 6, pp. 7700-7712, 2018, doi: 10.1109/ACCESS.2018.2803446.

[9] Fernández Maimó, L., Huertas Celdrán, A., Gil Pérez, M. *et al.* “Dynamic management of a deep learning-based anomaly detection system for 5G networks”. *J Ambient Intell Human Comput* 10, 3083–3097, 2019

[10] M. Doan and Z. Zhang, "Deep Learning in 5G Wireless Networks - Anomaly Detections," *2020 29th Wireless and Optical Communications Conference (WOCC)*, Newark, NJ, USA, 2020

[11] Y. Yuan, J. Yang, R. Duan, I. Chih-Lin, and J. Huang, "Anomaly Detection and Root Cause Analysis Enabled by Artificial Intelligence," *2020 IEEE Globecom Workshops (GC Wkshps*, Taipei, Taiwan, 2020

[12] Y. Khettab, M. Bagaa, D. L. C. Dutra, T. Taleb and N. Toumi, "Virtual security as a service for 5G verticals," *2018 IEEE Wireless Communications and Networking Conference (WCNC)*, Barcelona, Spain, 2018, pp. 1-6, doi: 10.1109/WCNC.2018.8377298.

[13] B. Weinger, J. Kim, A. Sim, M. Nakashima, N. Moustafa, K. J. Wu, “Enhancing IoT anomaly detection performance for federated learning”, Digital Communications and Networks,Volume 8, Issue 3, 2022

[14] A. Daly, ”The Legality of Deep Packet Inspection”,International Journal of Communications Law & Policy, No. 14, 2011

[15] L. F. Maimó, F. J. G. Clemente, M. G. Pérez, and G. M. Pérez, "On the performance of a deep learning-based anomaly detection system for 5G networks," *2017 IEEE SmartWorld, Ubiquitous Intelligence & Computing, Advanced & Trusted Computed, Scalable Computing & Communications, Cloud & Big Data Computing, Internet of People and Smart City Innovation (SmartWorld/SCALCOM/UIC/ATC/CBDCom/IOP/SCI)*, San Francisco, CA, USA, 2017

[16] M. Rodríguez, D. P. Tobón, D. Múnera, “Anomaly classification in industrial Internet of things: A review”, Intelligent Systems with Applications,Volume 18, 2023

[17] M. Zolanvari, A. Ghubaish and R. Jain, "ADDAI: Anomaly Detection using Distributed AI," *2021 IEEE International Conference on Networking, Sensing and Control (ICNSC)*, Xiamen, China, 2021

[18] Y. Wu, H. -N. Dai and H. Tang, "Graph Neural Networks for Anomaly Detection in Industrial Internet of Things," in *IEEE Internet of Things Journal*, vol. 9, no. 12, pp. 9214-9231, 15 June15, 2022

[19] M. Zolanvari, A. Ghubaish and R. Jain, "ADDAI: Anomaly Detection using Distributed AI," *2021 IEEE International Conference on Networking, Sensing and Control (ICNSC)*, Xiamen, China, 2021

[20] Z. Jadidi *et al*., "Security of Machine Learning-Based Anomaly Detection in Cyber Physical Systems," *2022 International Conference on Computer Communications and Networks (ICCCN)*, Honolulu, HI, USA, 2022

[21] P. Illy and G. Kaddoum, "A Collaborative DNN-Based Low-Latency IDPS for Mission-Critical Smart Factory Networks," in *IEEE Access*, vol. 11, pp. 96317-96329, 2023

[22] H.A. Alameddine, T. Madi, A. Boukhtouta, “How proactive anomaly detection secures 5G networks”, https://www.ericsson.com/en/blog/2021/8/proactive-anomaly-detection, 2021

[23] G. Lee, J. Seo and D. -k. Kim, "An Approach to Mitigate Wormhole Attack in Wireless Ad Hoc Networks," *2008 International Conference on Information Security and Assurance (isa 2008), Busan, Korea (South), 2008, pp. 220-225, doi: 10.1109/ISA.2008.44.*
[24] Correa, C., Robin, J., Mazo, R., Abreu, S. (2022). Intelligent Decision Support for Cybersecurity Incident Response Teams: Autonomic Architecture and Mitigation Search. In: Luo, B., Mosbah, M., Cuppens, F., Ben Othmane, L., Cuppens, N., Kallel, S. (eds) Risks and Security of Internet and Systems. CRiSIS 2021. Lecture Notes in Computer Science, vol 13204. Springer, Cham.
[25] Qazi Mamoon Ashraf, Mohamed Hadi Habaebi, Autonomic schemes for threat mitigation in Internet of Things, Journal of Network and Computer Applications,Volume 49, 2015
[26] C. Rouff, L. Watkins, R. Sterritt and S. Hariri, "SoK: Autonomic Cybersecurity - Securing Future Disruptive Technologies," *2021 IEEE International Conference on Cyber Security and Resilience (CSR), Rhodes, Greece, 2021*
[27] G. Gonzalez-Granadillo, J. Garcia-Alfaro, H. Debar, A polytope-based approach to measure the impact of events against critical infrastructures,Journal of Computer and System Sciences,Volume 83, Issue 1, 2017
[28] E. Doynikova and I. Kotenko, "The Multi-Layer Graph Based Technique for Proactive Automatic Response Against Cyber Attacks," *2018 26th Euromicro International Conference on Parallel, Distributed and Network-based Processing (PDP), Cambridge, UK, 2018*
[29] B. Fila and W. Wideł, "Exploiting attack–defense trees to find an optimal set of countermeasures," *2020 IEEE 33rd Computer Security Foundations Symposium (CSF), Boston, MA, USA, 2020*
[30] Correa, C., Robin, J., Mazo, R., Abreu, S. (2022). Intelligent Decision Support for Cybersecurity Incident Response Teams: Autonomic Architecture and Mitigation Search. In: Luo, B., Mosbah, M., Cuppens, F., Ben Othmane, L., Cuppens, N., Kallel, S. (eds) Risks and Security of Internet and Systems. CRiSIS 2021. Lecture Notes in Computer Science, vol 13204. Springer, Cham.
[31] DeMedeiros K, Hendawi A, Alvarez M., "A Survey of AI-Based Anomaly Detection in IoT and Sensor Networks" in *Sensors*, vol 23, no 3, 2023.
[32] TR IEC/TR 62443-3-1 "Industrial communication networks – Network and system security –Part 3-1 : Security technologies for industrial automation and control systems", 2009
[33] Industrie 4.0 Security Guidelines Recommendations for action
[34] ISO/IEC 27001:2013 Information technology -- Security techniques -- Information security management systems – Requirements
[35] ISO/IEC 27002:2013 Information technology -- Security techniques -- Code of practice for information security controls
[36] IEC 62443-3-3:2013 System security requirements and security levels
[37] IEC 62443-2-1:2010 Establishing an industrial automation and control system security program
[38] GSMA CLP.12 IoT Security Guidelines for IoT Service Ecosystems
[39] GSMA CLP.13 IoT Security Guidelines for Endpoint Ecosystems
[40] Industrial Security: Applying IoT Security Controls on the Industrial Plant Floor
[41] Smarter Security for Manufacturing in The Industry 4.0 Era: Industry 4.0 Cyber Resilience for the Manufacturing of the Future
[42] IEC 62443-2-1:2010 Establishing an industrial automation and control system security program
[43] Muniz, J., McIntyre, G., & AlFardan, N. (2015). *Security operations center: Building, operating, and maintaining your SOC. Cisco Press. ISO 690*
[44] Mughal, A. A. (2022). Building and Securing the Modern Security Operations Center (SOC). *International Journal of Business Intelligence and Big Data Analytics*, *5*(1), 1–15. Retrieved from https://research.tensorgate.org/index.php/IJBIBDA/article/view/21
[45] Cinque, M., Cotroneo, D., & Pecchia, A. (2018, October). Challenges and directions in security information and event management (SIEM). In *2018 IEEE International Symposium on Software Reliability Engineering Workshops (ISSREW)* (pp. 95-99). IEEE.
[46] Sridharan, A., & Kanchana, V. (2022, November). SIEM integration with SOAR. In *2022 International Conference on Futuristic Technologies (INCOFT)* (pp. 1-6). IEEE

[47] Kinyua, J., & Awuah, L. (2021). AI/ML in Security Orchestration, Automation and Response: Future Research Directions. Intelligent Automation & Soft Computing, 28(2).
[48] Tashfeen, M. T. A. (2023, July). Building blocks of incident response: Security operation centers. In *AIP Conference Proceedings* (Vol. 2814, No. 1). AIP Publishing.
[49] Jayasree Sengupta, Sushmita Ruj, Sipra Das Bit, A Comprehensive Survey on Attacks, Security Issues and Blockchain Solutions for IoT and IIoT, Journal of Network and Computer Applications, Volume 149,2020, 102481
[50] K. Tsiknas, D. Taketzis, K. Demertzis, and C. Skianis, "Cyber Threats to Industrial IoT: A Survey on Attacks and Countermeasures," *IoT, vol. 2, no. 1, pp. 163–186, Mar. 2021*
[51] J. P. Mohan, N. Sugunaraj and P. Ranganathan, "Cyber Security Threats for 5G Networks," *2022 IEEE International Conference on Electro Information Technology (eIT)*, Mankato, MN, USA, 2022
[52] Hasan, M.K., et al.: A review on security threats, vulnerabilities, and counter measures of 5G enabled Internet-of-Medical-Things. *IET Commun.* 16, 421–432 (2022).
[53] Silpa, C., Niranjana, G., Ramani, K. (2022). Securing Data from Active Attacks in IoT: An Extensive Study. In: Manogaran, G., Shanthini, A., Vadivu, G. (eds) Proceedings of International Conference on Deep Learning, Computing, and Intelligence. Advances in Intelligent Systems and Computing, vol 1396. Springer
[54] European union agency for cybersecurity, ENISA threat landscape for 5G Networks
[55] J. D. Day and H. Zimmermann, "The OSI reference model," in Proceedings of the IEEE, vol. 71, no. 12, pp. 1334-1340, Dec. 1983, doi: 10.1109/PROC.1983.12775.
[56] https://wazuh.com/
[57] https://kubernetes.io/en/docs/concepts/overview/what-is-kubernetes/
[58] https://attack.mitre.org/
[59] https://thehive-project.org/
[60] https://thehive-project.github.io/Cortex-Analyzers/analyzers/OpenCTI/
[61] https://github.com/OpenCTI-Platform/opencti
[62] https://docs.thehive-project.org/cortex/
[63] Sehan Samarakoon et al., December 2, 2022, "5G-NIDD: A Comprehensive Network Intrusion Detection Dataset Generated over 5G Wireless Network", IEEE Dataport
[64] X. Xu, J. Li, Y. Yang and F. Shen, "Toward Effective Intrusion Detection Using Log-Cosh Conditional Variational Autoencoder," in *IEEE Internet of Things Journal*, vol. 8, no. 8, 2021
[65] Vinyals, O., Fortunato, M., Jaitly, N., 2015. Pointer networks, in: Advances in Neural Information Processing Systems 28, pp. 2692–2700
[66] Mpatziakas A., Drosou A., Papadopoulos S., Tzovaras D., "IoT threat mitigation engine empowered by artificial intelligence multi-objective optimization", Journal of Network and Computer Applications, Volume 203,2022
[67] Q. Zhang and H. Li, "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition," in *IEEE Transactions on Evolutionary Computation, vol. 11, no. 6, pp. 712-731, Dec. 2007*
[68] Sutskever, I., Vinyals, O., Le, Q.V., 2014. Sequence to sequence learning with neural networks. Advances in neural information systems 27, 3104– 31112
[69] X. Zhang and J. You, "A Gated Dilated Causal Convolution Based Encoder-Decoder for Network Traffic Forecasting," in *IEEE Access*, vol. 8, pp. 6087-6097, 2020, doi: 10.1109/ACCESS.2019.2963449
[70] B. Shaojie, J. Z. Kolter, and V. Koltun. "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling." *arXiv preprint arXiv:1803.01271* (2018).
[71] Lara-Benítez, P.; Carranza-García, M.; Luna-Romera, J.M.; Riquelme, J.C. Temporal Convolutional Networks Applied to Energy-Related Time Series Forecasting. *Appl. Sci.* 2020, *10, 2322*
[72] Sehan Samarakoon, Yushan Siriwardhana, Pawani Porambage, Madhusanka Liyanage, Sang-Yoon Chang, Jinoh Kim, Jonghyun Kim, Mika Ylianttila, December 2, 2022, "5G-NIDD: A Comprehensive Network Intrusion Detection Dataset Generated over 5G Wireless Network", IEEE Dataport,

[73] A. Alshamrani, S. Myneni, A. Chowdhary and D. Huang, "A Survey on Advanced Persistent Threats: Techniques, Solutions, Challenges, and Research Opportunities," in *IEEE Communications Surveys & Tutorials*, vol. 21, no. 2, pp. 1851-1877, Secondquarter 2019

[74] Stellios, I., Kotzanikolaou, P., Psarakis, M. (2019). Advanced Persistent Threats and Zero-Day Exploits in Industrial Internet of Things. In: Alcaraz, C. (eds) Security and Privacy Trends in the Industrial Internet of Things. Advanced Sciences and Technologies for Security Applications. Springer, Cham

[75] C. Coldwell *et al*., "Machine Learning 5G Attack Detection in Programmable Logic," *2022 IEEE Globecom Workshops (GC Wkshps)*, Rio de Janeiro, Brazil, 2022, pp. 1365-1370

[76] Rui Pinto, March 24, 2020, "M2M using OPC UA ", IEEE Dataport, doi: https://dx.doi.org/10.21227/ychv-6c68.

[77] Benaddi H, Jouhari M, Ibrahimi K, Ben Othman J, Amhoud EM. Anomaly Detection in Industrial IoT Using Distributional Reinforcement Learning and Generative Adversarial Networks. *Sensors*. 2022; 22(21):8085. https://doi.org/10.3390/s22218085

[78] Neto ECP, Dadkhah S, Ferreira R, Zohourian A, Lu R, Ghorbani AA. CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment. *Sensors*. 2023; 23(13):5941.

[79] M. S. Elsayed, N. -A. Le-Khac and A. D. Jurcut, "InSDN: A Novel SDN Intrusion Dataset," in *IEEE Access*, vol. 8, pp. 165263-165284, 2020,

[80] A. Dijk, "Detection of Advanced Persistent Threats using Artificial Intelligence for Deep Packet Inspection," *2021 IEEE International Conference on Big Data (Big Data)*, Orlando, FL, USA, 2021, pp. 2092-2097

[81] Fether, Scott D. "PCAP Next Generation: Is Your Sniffer Up to Snuff?.", 2018 *ISSA Journal* 16.7

[82] Wireshark documentation , https://wiki.wireshark.org/ProtocolReference , accessed 29/10/2023

[83] TCPdump documentation, https://www.tcpdump.org/manpages/tcpdump.1.html, accessed 29/10/2023

Appendix A: The OSI reference model

The OSI reference model is mentioned in section 2.1. The OSI (Open Systems Interconnection) model is a conceptual framework used to understand and design computer networks. It divides network communication into seven abstraction layers, each responsible for specific functions[55]. Here's a brief overview of the OSI model from the lowest to the highest layer:

1. **Physical Layer:** Deals with the physical connection between devices. It defines aspects like cables, connectors, and the electrical signals used for transmission.
2. **Data Link Layer:** Responsible for the reliable transmission of data between adjacent network nodes. It includes error detection and correction, as well as techniques for flow control.
3. **Network Layer:** Manages the routing of data packets between devices on different networks. It addresses issues such as logical addressing, routing, and packet forwarding.
4. **Transport Layer:** Ensures end-to-end communication, providing error detection, correction, and flow control. It can establish, maintain, and terminate connections.
5. **Session Layer:** Manages sessions or dialogues between applications. It establishes, maintains, and terminates connections, allowing for full-duplex or half-duplex communication.
6. **Presentation Layer:** Responsible for data translation, encryption, and compression, ensuring that data is presented in a readable format between applications.
7. **Application Layer:** The top layer, it directly interacts with end-user applications and provides network services like email, file transfer, and network browsing.

Appendix B: Measures utilized to assess binary classification

The following appendix contains the measures utilized to assess the success of the binary classification performed by the Ultralight Anomaly Detection submodule presented in section 3.2.1.1
The following table is called a confusion matrix and the values of it allow the calculation of multiple metrics.

Table 3 Confusion Matrix

		Actual Values	
		Positive (Benign Traffic)	Negative (Anomalous Traffic)
Predicted Values	Positive (Benign Traffic)	TP	FP
	Negative (Anomalous Traffic)	FN	TN

- True Positive Rate (TPR) or Hit Rate or Recall or Sensitivity = TP / (TP + FN)
- False Positive Rate (FPR) or False Alarm Rate = 1 - Specificity = 1 - (TN / (TN + FP))
- Accuracy = (TP + TN) / (TP + TN + FP + FN)
- Error Rate = 1 – accuracy or (FP + FN) / (TP + TN + FP + FN)
- Precision = TP / (TP + FP)
- F-measure: 2 / ((1 / Precision) + (1 / Recall))
- ROC (Receiver Operating Characteristics) = plot of FPR vs TPR
- AUC (Area Under the Curve)