Industrial Cyber Shield: Protecting Industrial Networks in the Age of Industry 4.0

Gabriele Peluzzi[1], Giorgia Mannarino[1], Bianca Mazzà[1], Stefano Perone[1], Ernesto Del Prete[2], Simone Guarino[1]

[1] Unit of Automatic Control
Department of Engineering
Campus Bio-Medico University of Rome
Via Alvaro del Portillo 21, 00128 Roma, Italy

[2] INAIL
Department of Technological Innovations and Safety of Plants, Products and Anthropic Settlements, Italy

Email: {gabriele.peluzzi; giorgia.mannarino; bianca.mazza; s.perone; s.guarino}@unicampus.it; e.delprete@inail.it

***Abstract*—The transition toward Industry 4.0 exposes SCADA systems to significant cyber-physical risks due to increased remote connectivity. The Industrial Cyber Shield (ICS) project, funded by the National Institute for Insurance against Accidents at Work (INAIL), responds to this challenge through an intrusion response system that is composed of two main modules: a centralized Orchestrator and multiple distributed Shield modules. The Shields are deployed inline to protect Programmable Logic Controllers (PLCs), employing Snort for real-time network analysis and recurrent neural networks for physical process anomaly monitoring. Detected events are transmitted via a Kafka broker to the Orchestrator, which provides automated response instructions in real-time. Experimental validation on a water distribution testbed demonstrates that the architecture introduces negligible latency (less than 1 ms) and maintains limited hardware resource consumption. The system successfully neutralizes simulated attacks while preserving the stringent timing requirements of industrial communications, ensuring both operational transparency and plant safety.**

***Index Terms*—Industry 4.0, SCADA Systems, Cybersecurity, Intrusion Detection Systems, Anomaly Detection**

I. Introduction

The digitization of industrial processes has significantly improved efficiency but exposed critical infrastructures to sophisticated cyber threats, creating complex networks that are inherently more vulnerable [1]. In Industrial Cyber-Physical Systems (ICPSs), traditional isolation paradigms based on the concept of *security-by-obscurity* have been progressively abandoned in favor of open and heterogeneous communication standards, enabling remote connectivity through the Internet and tighter integration with corporate networks [2]. As a consequence, Supervisory Control and Data Acquisition (SCADA) systems, originally designed as closed and trusted environments, can now be reached through legacy industrial protocols that were not conceived with security in mind, thus opening new attack vectors that directly impact both business continuity and worker safety [3], [4]. Legacy SCADA devices often lack modern security safeguards, making them susceptible to attacks that can manipulate physical assets through communication channels and Programmable Logic Controllers (PLC) [5]. A comprehensive analysis of SCADA vulnerabilities shows that threats originate from various layers, including application servers, network protocols, and field devices like RTUs and PLCs [6].

Conventional cybersecurity solutions for industrial environments mainly rely on perimeter defenses, static firewalling, and centralized Intrusion Detection Systems (IDSs). While such approaches remain essential, they often struggle to provide timely and context-aware reactions in modern distributed SCADA networks [7]. The distributed and heterogeneous nature of ICPSs calls for protection paradigms that combine local monitoring and mitigation capabilities — characterized by low latency and high operational transparency — with centralized coordination for advanced incident analysis and response orchestration.

In this regard, the scientific literature has extensively investigated anomaly detection techniques for industrial processes, with a strong focus on data-driven approaches based on machine learning and deep learning. Several works target the detection of cyber and physical anomalies using multivariate time-series models and unsupervised learning, but they typically operate on offline datasets and rarely discuss the real-time performance of the underlying data acquisition and processing pipelines [8], [9]. Moreover, despite the growing interest in ICPS security, automatic response mechanisms remain largely underexplored due to the high risks associated with enforcing countermeasures in complex industrial plants, where incorrect actions may lead to production losses, environmental damage or even human casualties [10].

To address these gaps, the Italian national project Industrial Cyber Shield (ICS), funded by the National Institute for Insurance against Accidents at Work (INAIL), aims to design an intrusion response system that assists operators and enables automated deployment when conditions are critical [11]. A key enabling component of such a system is an effective Anomaly Detection System, capable of jointly processing cyber and physical data coming from the SCADA network in real time. In our previous work [12], we presented a real-time data cap-

This work was supported by Agenzia per la cybersicurezza nazionale under the programme for promotion of XL cycle PhD research in cybersecurity – C83C24000790001. The views expressed are those of the authors and do not represent the funding institution.

ture probe for anomaly detection in ICPSs, implemented on an Odroid platform and validated on a water distribution testbed. That study focused on guaranteeing real-time performance for data sniffing and feature extraction, analyzing metrics such as CPU usage, memory usage, and system wait time to ensure that anomaly detection could be performed without violating the polling deadlines of the Human-Machine Interface (HMI).

Building on these results, the ICS project introduces a dual cyber-physical detection framework that integrates data-driven process-monitoring techniques with rule-based mechanisms for network security; the two branches are decoupled at the Shield to preserve real-time guarantees on the inline path and converge at the Orchestrator, which time-aligns Snort alerts and LSTM residuals per Shield to enable cross-domain correlation. On the physical side, Long Short-Term Memory (LSTM) recurrent neural networks are employed to monitor industrial processes and detect deviations from nominal behavior. Concurrently, on the cyber side, the system integrates Snort[1], a widely adopted rule-based IDS, to perform rule-based recognition of known attacks directly at the network level. LSTM neural networks are particularly effective for monitoring industrial multivariate time series data, as they can learn the predictable and regular nature of the physical process to detect anomalies with high precision [13], [14]. The availability of realistic datasets, such as the hardware-in-the-loop water distribution testbed (WDT) for cyber-physical security testing [15], plays a crucial role in designing, training, and validating such algorithms on scenarios that closely resemble real industrial plants.

This paper presents the ICS system within this context. Developed as part of the INAIL BRIC 2022 program, the project is carried out by a research consortium including Campus Bio-Medico University of Rome and INAIL. The ICS architecture is based on a two-level edge computing model specifically designed to detect and respond to cyber threats and process anomalies in networked industrial environments. At the field level, distributed Shield devices are deployed in inline mode between PLCs and the rest of the SCADA network, ensuring transparent and low-latency communication while enabling both rule-based detection via Snort IDS and anomaly-based detection via LSTM neural networks on process data. At the centralized level, an Orchestrator coordinates event correlation, advanced analysis, and the activation of automatic mitigation responses across the Shield devices. Communication between the Shields and the Orchestrator is implemented through an Apache Kafka broker, protected by mutual TLS with X.509 certificates and per-Shield payload signing, so that origin and integrity of alerts and mitigation commands are verifiable end-to-end. Modbus/TCP traffic is instead left unencrypted, since retrofitting confidentiality into the legacy protocol is generally not feasible without breaking PLC compatibility [2], [6] — which is itself a key motivation for the inline Shield deployment.

The use of message-oriented middleware like Apache Kafka plays a key role in decoupling data producers and consumers, ensuring scalable, reliable, and real-time streaming processing in federated IoT environments [16], [17]. The ICS system pursues a dual objective: preventing hazardous situations through operator guidance and automatically enforcing mitigation actions when required. In particular, the Shields leverage the `nftables` firewall to dynamically enforce filtering rules on their network bridge, selectively blocking malicious traffic before it reaches the PLCs, while leaving legitimate traffic unaffected. Automated incident response systems are increasingly necessary to defend PLCs against volumetric attacks, such as DoS, by executing pre-defined filtering rules at the network level [18]. Experimental validation on a water distribution testbed demonstrates that the proposed architecture introduces negligible latency (less than 1 ms) and maintains limited hardware resource consumption, thus preserving the stringent timing requirements of industrial communications. The system successfully neutralizes simulated attacks and reliably signals process anomalies, ensuring both operational transparency and plant safety.

The remainder of the paper is organized as follows. Section II describes the ICS system architecture, the reference testbed, and the detection and response mechanisms implemented on the Shields and the Orchestrator. Section III presents the experimental results in terms of detection effectiveness, network latency, and resource usage. Finally, Section IV draws the main conclusions and outlines future research directions.

II. ICS System Architecture

A. Reference Testbed

The project was designed and implemented at the Campus Bio-Medico University of Rome on a water distribution testbed composed of 8 tanks, 6 pumps, 22 solenoid valves, 8 pressure sensors, and 4 flow sensors [15]. The physical process is controlled by 4 PLCs interconnected within a SCADA network using the Modbus TCP/IP communication protocol. Plant supervision is performed in real time via a central workstation equipped with an HMI and a Historian. The testbed and network scheme are shown in Figure 1.

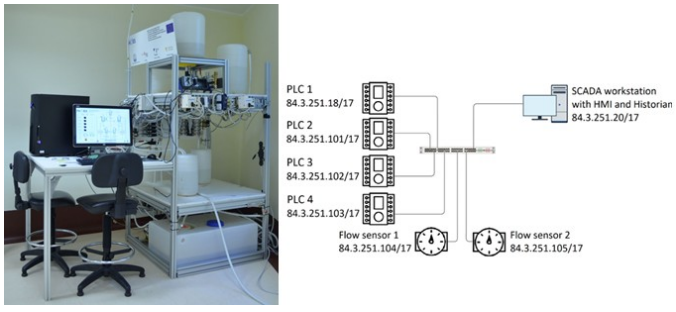

Fig. 1. Water distribution testbed at the Campus Bio-Medico University of Rome.

B. Architecture

The project architecture, shown in Figure 2, is characterized by a two-level system. The first, referred to as the field

[1] https://www.snort.org/

level, aims to protect the PLCs by detecting and potentially countering threats to plant security, such as process anomalies or cyber attacks. The second aims to coordinate and manage the proper operation of the field devices, referred to as Shields, by defining operating policies as well as the actions to be taken following the detection of attacks and/or process anomalies. To protect each of the 4 PLCs in the water testbed, 4 Odroid H4+ Shield devices have been deployed, each communicating with a single Orchestrator device via a Kafka broker.

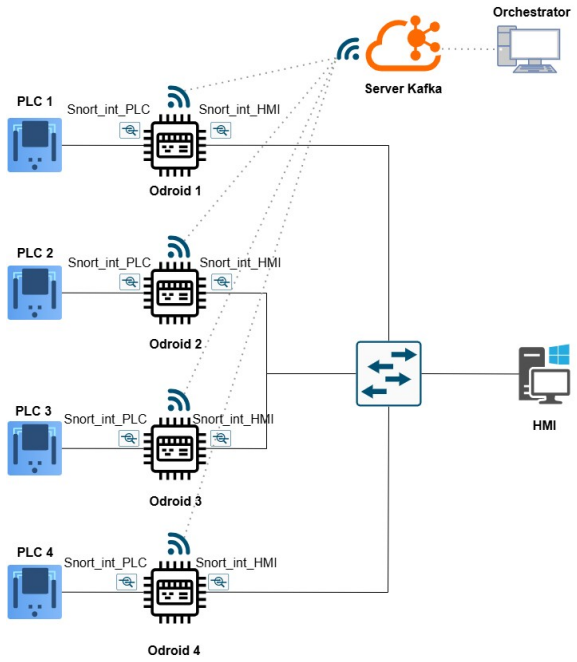

Fig. 2. ICS Architecture.

1) Field Device: Shield: Field devices are configured in inline mode to decouple individual PLCs from the rest of the SCADA network at the network level. To this end, each Shield establishes a network bridge between the two interfaces, ensuring on one hand proper communication between the PLC and the remaining SCADA network devices, and on the other, the ability to analyze and potentially filter network traffic if a cyber attack is recognized. The Shields were developed on Odroid H4+ devices, each equipped with 3 network interfaces: two interfaces in a bridge configuration between the PLC and the SCADA network, and a third wireless interface for communication with the Orchestrator device. The Shield device aims to (i) ensure transparent communication, (ii) recognize cyber attacks or process anomalies through real-time traffic analysis, and (iii) communicate events to the Orchestrator for immediate mitigation.

To ensure comprehensive monitoring, the system adopts two complementary strategies for the recognition of process anomalies and cyber attacks. Regarding process anomalies, the approach is based on LSTM recurrent neural networks aimed at detecting anomalies in the water tank filling and emptying process. Specifically, eight distinct networks were trained, one for each tank in the testbed, using historical datasets related to nominal filling and emptying cycles. This training allows the system to learn the plant's nominal behavior and identify significant deviations in real-time. During the operational phase, the Shield device continuously intercepts the content of Modbus response packets from the PLC to the HMI, extracting the current tank levels from the payload. These values are then compared with the predictions generated by the LSTM model: a significant deviation between the measured value and the expected one is flagged as a process anomaly. In Figure 3, an example of anomaly detection on tank 1 of the water testbed is shown.

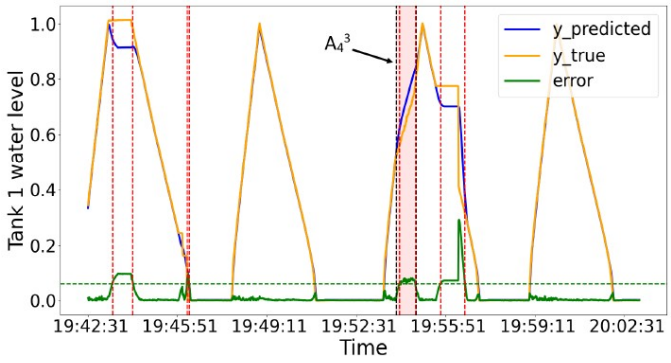

Fig. 3. Detection of a process anomaly on tank 1.

Cyber attack detection relies on Snort, configured in its IDS mode, which enables the integration of custom rules to identify known threats. Within the project context, specific rules have been defined to detect three main categories of attacks:

- Denial of Service (DoS), in its two variants: TCP flood and ICMP flood.
- Scanning, in its five variants: Null scan, FIN scan, SYN scan, XMAS scan, and ICMP scan.
- ARP poisoning.

The architecture involves running Snort instances on both network interfaces of the bridge (PLC side and SCADA side). This dual monitoring configuration is necessary for the precise localization of the attack origin, while also enabling verification of countermeasure effectiveness by observing whether malicious traffic is blocked before reaching the target device. Upon a positive detection, the Shield triggers the immediate activation of a local audio-visual alarm to alert on-site operators and simultaneously transmits an alert message to the centralized Orchestrator via the Kafka broker.

For the implementation of countermeasures, `nftables`, the modern Linux packet filtering firewall, was adopted, motivated by its capability to inspect traffic from Layer 2 (e.g., MAC address) up to Layers 3/4 (IP, protocols, TCP/UDP ports). Response actions consist of dynamic firewalling rules on the Shield's network bridge, intercepting malicious packets on the external interface before they reach the PLC. `nftables`, like any netfilter-based framework, performs no payload inspection above L4 and can be evaded by fragmentation, spoofing within the authorized range, or low-and-slow traffic; for this reason it is used here not as a self-sufficient

defense but as one enforcement layer of a defense-in-depth scheme, coupled with Snort on both bridge interfaces and with a hardened, single-purpose Shield OS. The specific filtering actions for each of the attacks detectable by Snort are listed below.

- **DoS/DDoS Attacks.** The response consists in limiting the network throughput for the affected protocol, either TCP (in the case of TCP flood) or ICMP (in the case of ICMP flood).
- **Scanning Attacks.** If the attacker's IP or MAC is not in the list of authorized SCADA addresses, the offending address notified by Snort is filtered; if both are authorized, no automatic action is taken (as it would disrupt legitimate devices) and the Orchestrator notifies the operator.
- **ARP Poisoning.** If the MAC reported by Snort is unauthorized, it is filtered; otherwise no action is taken on the Shield and the Orchestrator notifies the operator, following the same rationale as above.

Finally, should the Shield lose communication with the Orchestrator, it will execute a default policy that can be customized by the operator. Such policies may range from a simple notification to the operator to the forced interruption of the network bridge, resulting in the blocking of traffic.

2) Orchestration Device: Orchestrator: The Orchestrator performs three main functions:

- Distributing initial configuration files to each Shield to provide the correct information for attack detection.
- Receiving real-time attack notifications from the Shield devices.
- Issuing appropriate response actions to the Shields for network traffic filtering via the `nftables` framework.

As anticipated in Section I, communication between the Orchestrator and the Shields is enabled via a Kafka broker, with confidentiality and integrity guaranteed by mutual TLS and per-Shield payload signing. The adoption of Kafka, in particular, allows for information exchange based on specific topics. In this way, each Shield will communicate attacks and receive responses on its own specific topics, thereby ensuring that each device can access only the information relevant to it.

Regarding the transmission of the configuration file, each Shield receives the following information upon startup:

- The IP address of the PLC to be protected.
- The IP addresses and corresponding MAC addresses of the devices connected to the SCADA network.
- The maximum acceptable threshold for the packet-per-second rate.

Based on the received configuration information, each Shield initializes its local IDS rules to detect the three classes of attacks. When Snort identifies an attack, the information is transmitted to the Orchestrator, which, in response, sends specific commands to activate countermeasures. These commands are Base64-encoded Bash scripts that, once executed by the Shield, dynamically insert the appropriate `nftables` filtering rules to neutralize the specific detected threat. Simultaneous monitoring on the two bridge interfaces (towards the PLC and towards the SCADA network) enables the classification of the attack evolution into three distinct scenarios:

1) **Active unmitigated attack** (detected on both interfaces): indicates that malicious traffic is traversing the bridge and reaching the PLC. It is the initial critical condition that triggers the request for intervention.
2) **Successfully mitigated attack** (detected only on the external/SCADA interface): confirms that the firewalling rule is active and is blocking malicious traffic before it traverses the bridge, effectively protecting the PLC even though the attack persists on the external network.
3) **Terminated attack** (not detected on any interface): indicates the cessation of hostile activity. In this case, the Orchestrator can order the removal of blocking rules to restore normal operation.

The first scenario occurs when an attacker attempts to access the PLC network segment from the SCADA network. In the absence of adequate countermeasures on the bridge, the attack traverses it, generating the same Snort event on both monitored interfaces. Following detection, the Orchestrator sends the appropriate countermeasure to block the passage of the attack on the bridge, thereby preventing the attacker from reaching the PLC. Specifically, the Orchestrator will send a Base64-encoded JSON5 message containing the Bash code for inserting the `nftables` rule.

The second scenario occurs following the receipt and implementation of the countermeasure. If the attack is detected exclusively on the external interface (SCADA side) but not on the internal one, it indicates that the `nftables` rule is effectively blocking malicious traffic on the bridge. In this case, the Orchestrator notifies the operator that the attack is contained but still active on the external network, maintaining filtering pending its conclusion. If, however, the attack continues to be detected on both interfaces despite the countermeasure, it implies that the filtering is ineffective and the PLC remains exposed. The Orchestrator immediately escalates the event priority, alerting the operator to the persistence of a critical threat that is not being automatically managed.

When the attack disappears from both interfaces, it can be deduced that hostile activity has ceased. Consequently, the Orchestrator sends a rollback command to the Shield to remove the temporary filtering rules, restoring full bridge transparency and normal plant operation.

Finally, it should be noted that countermeasures have been defined only for cyber attacks and not for process anomalies. For the latter, management is entirely delegated to the human operator, who must physically intervene on the infrastructure; this design choice aims to prevent unsupervised automatic actions from inadvertently aggravating the plant's state or compromising physical safety.

III. Results

Experimental validation was conducted by deploying two Odroid H4+ Shields to protect the first two PLCs of the plant. Each device is interfaced with an Arduino board dedicated to

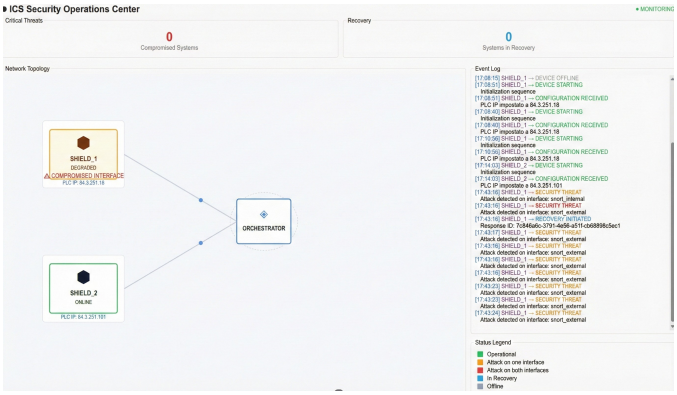

Fig. 4. Orchestrator dashboard reporting the detection of an attack against the first PLC.

driving two distinct audio-visual alarms, used to immediately distinguish between detections of cyber attacks and physical process anomalies.

During functional tests, attack scenarios were simulated using a Kali Linux virtual machine to generate the three classes of cyber attacks, while process anomalies were induced by physically manipulating the manual valves of the testbed. The results confirm the effectiveness of the system: in all evaluated scenarios, the Shields correctly identified the threats and executed the complete mitigation sequence. Specifically, an attack is initially detected on both network interfaces (SCADA and PLC sides). Upon receiving the alert, the Orchestrator responds by sending the necessary `nftables` countermeasures to block the malicious traffic. Once applied, the attack is detected exclusively on the external SCADA interface, confirming that the traffic is successfully blocked before reaching the PLC. Finally, when the attack ceases and is no longer detected on either interface, the Orchestrator issues a command to remove the temporary filtering rules, fully restoring the firewall to its normal transparent operation. Figure 4 shows the Orchestrator dashboard highlighting this correct operational sequence, from the initial distribution of configurations to the real-time detection and mitigation of an ongoing attack on the first Shield. Figure 5 shows the detection of process anomalies. The local dashboard, displayed directly on the Odroid screens, monitors the status of the five tanks managed by the first PLC. The example shows a forced anomaly on the first tank: the system reacts by highlighting the graph of the compromised level in red and simultaneously activating the dedicated alarm, thus ensuring immediate feedback to on-site operators.

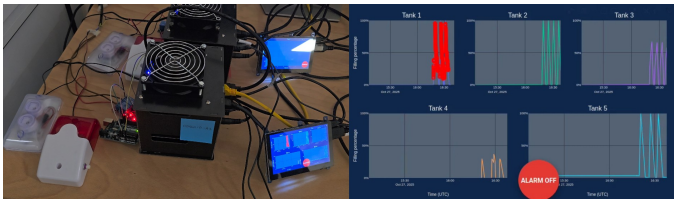

Fig. 5. Dashboard of the first Odroid reporting a detected anomaly on the first tank.

A. Network Performance Analysis

To assess the impact of introducing the Shield device on industrial communication, an in-depth analysis of Modbus TCP packet network latency (Round Trip Time) was conducted. Performance was compared in two configurations:

- *No Bridge (Baseline)*: direct connection between SCADA and PLC.
- *With Bridge (Shield)*: connection filtered through the Odroid device in transparent bridge mode.

Tests were performed by varying the polling time (query interval) of the HMI across five levels: 100 ms, 250 ms, 500 ms, 750 ms, and 1000 ms. For each configuration, packets were captured and analyzed for a duration of 600 seconds (10 minutes), ensuring a significant statistical sample (between 17,000 and 58,000 samples per test).

TABLE I
MODBUS TCP LATENCY COMPARISON (WITH BRIDGE VS NO BRIDGE)

Polling Time (ms)	**Avg Delay No Bridge (ms)**	**Avg Delay With Bridge (ms)**	**Δ Latency (ms)**
100	8.57 ($\sigma = 1.94$)	8.53 ($\sigma = 1.78$)	≈ 0.00
250	8.52 ($\sigma = 1.79$)	8.39 ($\sigma = 1.25$)	≈ 0.00
500	8.17 ($\sigma = 2.77$)	8.59 ($\sigma = 2.77$)	+0.42
750	8.14 ($\sigma = 3.32$)	8.57 ($\sigma = 2.74$)	+0.43
1000	7.87 ($\sigma = 3.28$)	8.53 ($\sigma = 3.27$)	+0.66

σ = Standard Deviation. Values based on 10-min capture.

The results, summarized in Table I, show that the overhead introduced by the Shield remains limited. Comparing the average latency values, it is observed that the difference between the direct configuration and the Shield configuration is always less than 1 ms. In particular, for high polling frequencies (100 ms and 250 ms), average latencies appear statistically comparable (differences on the order of hundredths of a millisecond, attributable to normal network variance). In slower polling configurations (500-1000 ms), the average latency increase stands at around 0.4-0.6 ms. This slight deviation is largely attributable to the inherent randomness of network traffic and the operating system's scheduling dynamics; with less frequent packet arrivals, the physiological network jitter and processing variance become proportionally more evident compared to continuous, high-frequency traffic scenarios.

B. System Resource Analysis

In parallel with the network analysis, the hardware resource consumption (CPU and RAM) of the Shield device was monitored to evaluate the computational cost of traffic inspection. Monitoring focused on the `packet_capture` process, responsible for packet interception and analysis. Table II directly compares the values recorded in the two operating configurations (i.e., *With Bridge* vs. *No Bridge*).

- **CPU Usage:** The results indicate that the additional overhead primarily affects CPU utilization. In the *No Bridge* scenario, the capture process is inactive (0%). With the active Shield, usage scales proportionally to

TABLE II
SYSTEM RESOURCE COMPARISON (WITH BRIDGE VS NO BRIDGE)

Polling (ms)	Avg CPU Usage (%)		Avg Mem Usage (%)		Avg RSS (MB)		Load Avg (1 min)	
	With Bridge	No Bridge	With Bridge	No Bridge	With Bridge	No Bridge	With Bridge	No Bridge
100	17.73	0.00	2.45	2.44	386.06	384.43	0.96	0.00
250	19.19	0.00	2.42	2.44	381.45	384.43	0.51	0.07
500	10.56	0.00	2.45	2.44	385.05	384.43	0.03	0.02
750	6.63	0.00	2.45	2.44	386.12	384.43	0.25	0.01
1000	3.98	0.00	2.45	2.44	385.81	384.43	0.12	0.01

traffic, rising from 3.98% (1000 ms) to a peak of 19.19% (250 ms) in the case of the most aggressive polling.

- **Load Average:** The impact on system load (1-min Load Average), a metric representing the average number of processes either executing or waiting for CPU resources, remains limited. Even under maximum stress (100 ms), the value stands at 0.96, remaining below the critical saturation threshold (1.0 per core), whereas in the scenario without protection, it is virtually zero.
- **Memory (RAM):** The impact on memory is negligible. Both the percentage usage ($\sim$2.45%) and the Resident Set Size (RSS $\sim$385 MB) remain virtually identical in the two configurations. This confirms that resource allocation for analysis models is static and does not depend on the volume of analyzed traffic.

These results confirm that the Odroid H4+ is properly dimensioned to support the Industrial Cyber Shield, ensuring plant security and real-time inspection without bottlenecks. The system preserves operational continuity with a negligible impact on memory and a well-managed CPU load.

IV. CONCLUSIONS

The Industrial Cyber Shield project has demonstrated how a two-level architectural approach, based on local field devices (Shields) integrated in inline mode with centralized coordination (Orchestrator), represents a viable solution for the protection of modern SCADA networks. The implementation of hybrid detection, combining rule-based analysis via Snort and anomaly-based analysis via LSTM neural networks, enables the identification of both known cyber attacks and process anomalies. The results of the experimental validation on a water distribution testbed confirm the system's effectiveness: all simulated attack scenarios were successfully mitigated. Performance analysis demonstrated negligible latency overhead (< 0.7 ms worst case), modest CPU usage ($< 20\%$ at peak) and virtually no memory overhead. Unlike centralized IDS-only solutions [2], [6] that lack automatic response, PLC-embedded approaches [18] constrained by controller resources, and offline cyber-physical fusion frameworks [11], [13], ICS jointly delivers edge cyber-physical detection, sub-millisecond inline overhead and automated reversible enforcement on a Modbus testbed. Future work will move from event-level correlation to a tighter cyber-physical fusion based on joint inference over Snort features and LSTM residuals, to better detect process-aware attacks with nominal network footprint.

REFERENCES

[1] B. Häckel, F. Hänsch, M. Hertel, and J. Übelhör, "Assessing it availability risks in smart factory networks," *Business Research*, vol. 12, no. 2, pp. 523–558, 2019.

[2] D. Pliatsios, P. Sarigiannidis, T. Lagkas, and A. Sarigiannidis, "A survey on scada systems: Secure protocols, incidents, threats and tactics," *IEEE Communications Surveys & Tutorials*, vol. 22, no. 3, pp. 1942–1976, 2020.

[3] L. Cazorla, C. Alcaraz, and J. López, "Cyber stealth attacks in critical information infrastructures," *IEEE Systems Journal*, vol. 12, no. 2, pp. 1778–1792, 2018.

[4] H.-N. Wu, H.-N. Dai, and H. Wang, "Convergence of blockchain and edge computing for secure and scalable iiot critical infrastructures in industry 4.0," *IEEE Internet of Things Journal*, vol. 8, no. 4, pp. 2300–2317, 2021.

[5] E. Wai and C. Lee, "Seamless industry 4.0 integration: A multilayered cyber-security framework for resilient scada deployments in cpps," *Applied Sciences*, vol. 13, no. 21, p. 12008, 2023.

[6] M. Alanazi, A. Mahmood, and M. J. M. Chowdhury, "Scada vulnerabilities and attacks: A review of the state-of-the-art and open issues," *Computers & security*, vol. 125, p. 103028, 2023.

[7] P.-Y. Chen, S. Yang, and J. A. McCann, "Distributed real-time anomaly detection in networked industrial sensing systems," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 6, pp. 3832–3842, 2015.

[8] L. Liu, M. Hu, C. Kang, and X. Li, "Unsupervised anomaly detection for network data streams in industrial control systems," *Information*, vol. 11, no. 2, p. 105, 2020.

[9] H. R. Nizam, S. Zafar, Z. Lv, F. Wang, and X. Hu, "Real-time deep anomaly detection framework for multivariate time-series data in industrial iot," *IEEE Sensors Journal*, vol. 22, no. 23, pp. 22 836–22 849, 2022.

[10] M. Barzegar and M. Shajari, "Attack scenario reconstruction using intrusion semantics," *Expert Systems with Applications*, vol. 108, pp. 119–133, 2018.

[11] S. Guarino, F. Vitale, F. Flammini, L. Faramondi, N. Mazzocca, and R. Setola, "A two-level fusion framework for cyber-physical anomaly detection," *IEEE Transactions on Industrial Cyber-Physical Systems*, vol. 2, pp. 1–13, 2023.

[12] R. F. S. Buttiglione, A. Gallo, S. Perone, E. Del Prete, and R. Setola, "A real-time data capture probe for anomaly detection in industrial cyber-physical systems," pp. 707–712, 2025.

[13] L. Faramondi, F. Flammini, S. Guarino, and R. Setola, "A hybrid behavior-and bayesian network-based framework for cyber–physical anomaly detection," *Computers and Electrical Engineering*, vol. 112, p. 108988, 2023.

[14] P. Filonov, A. Lavrentyev, and A. Vorontsov, "Multivariate industrial time series with cyber-attack simulation: Fault detection using an lstm-based predictive data model," *arXiv preprint arXiv:1612.06676*, 2016.

[15] L. Faramondi, F. Flammini, S. Guarino, and R. Setola, "A hardware-in-the-loop water distribution testbed dataset for cyber-physical security testing," *IEEE Access*, vol. 9, pp. 122 385–122 396, 2021.

[16] P. Bellavista, R. Della Penna, L. Foschini, and D. Scotece, "Machine learning for predictive diagnostics at the edge: An iiot practical example," pp. 1–7, 2020.

[17] J. Sychowiec and Z. Zieliński, "An experimental framework for secure and reliable data streams distribution in federated iot environments," pp. 769–780, 2023.

[18] R. D. Werth, T. Morris *et al.*, "A multi-layered embedded intrusion detection framework for programmable logic controllers," *arXiv preprint arXiv:2510.07171*, 2025.