Internet of Things 37 (2026) 101915

Contents lists available at ScienceDirect

Internet of Things

journal homepage: www.elsevier.com/locate/iot

Anomaly detection of cyber threats in industrial IoT networks via hybrid digital twins and continual learning

Andrea Melis [a,*], Andrea Piroddi [a], Chan-Tong Lam [b], Giovanni Pau [a], Roberto Girau [a]

[a] Department of Computer Science and Engineering, University of Bologna, Via Zamboni 33, Bologna, Italy
[b] Faculty of Applied Sciences, University of Macau, Macao, China

ARTICLE INFO

Keywords:
Industrial Internet of Things
Digital twin
Anomaly detection
Security
Neural network
Hardware in the loop

ABSTRACT

The Industrial Internet of Things (IIoT) is increasingly exposed to cyber threats due to its tight integration of operational technology and digital connectivity. Traditional intrusion detection systems (IDSs) often struggle with adaptability, false positives, and operational scalability in dynamic, non-stationary environments. This paper proposes a cyber threat detection framework that integrates hybrid digital twins (DTs) with continual learning to enable reliable and adaptive intrusion detection in realistic IIoT settings. The hybrid DTs act as local mirrors of IIoT devices, preserving sensitive data close to the source while supporting controlled validation of firmware updates and configuration changes. The continual learning mechanism enables the detection model to incrementally adapt to evolving traffic patterns and emerging attacks, mitigating catastrophic forgetting without requiring repeated offline retraining. Experimental validation on benchmark datasets and real IIoT traffic shows that the proposed DT-enabled framework supports stable detection performance over time under bounded memory and incremental update constraints, reflecting realistic deployment conditions. The proposed architecture highlights a practical trade-off between offline optimality and online adaptability, offering a robust, scalable solution for securing IIoT infrastructure that balances continuous operation, reliability, and controlled adaptation.

1. Introduction

The Industrial Internet of Things (IIoT) is transforming industrial systems by enabling interconnected devices and distributed components to communicate and collaborate efficiently. However, this increased connectivity comes at a cost. Securing IIoT environments remains particularly challenging due to strong device heterogeneity, which hinders uniform protection across networks, and the widespread presence of legacy systems that were never designed with cybersecurity requirements in mind, leaving them vulnerable to modern attack vectors [1]. As the number of connected devices grows, effective intrusion detection systems (IDSs) become essential for identifying and responding to cyber threats in real time. Testing and validating such systems in industrial environments, however, is inherently difficult. Accurately simulating industrial devices and generating representative traffic is often infeasible, as the diversity of industrial protocols, firmware, and architectures limits the availability of comprehensive and realistic datasets. Moreover, evaluating attacks and defense mechanisms directly on a live industrial plant is rarely acceptable, as these environments are safety-critical and cannot tolerate service interruptions or unstable behavior. To address these challenges, the concept of the digital

* Corresponding author.
E-mail address: a.melis@unibo.it (A. Melis).

https://doi.org/10.1016/j.iot.2026.101915
Received 16 September 2025; Received in revised form 12 January 2026; Accepted 26 February 2026
Available online 28 February 2026
2542-6605/© 2026 The Author(s). Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

twin has become increasingly relevant in industrial contexts. Digital twins enable the creation of virtual counterparts of physical assets, supporting monitoring, analysis, and controlled experimentation without impacting ongoing operations. In IIoT scenarios, digital twins allow security mechanisms to be tested under realistic conditions, facilitate the safe evaluation of configuration changes and firmware updates, and enable the exploration of hazardous scenarios in a risk-free environment. Despite these advantages, the training and maintenance of effective IDS models remain critical challenges. Industrial systems exhibit highly dynamic operational behavior, with traffic patterns evolving due to workload variations, production cycles, maintenance activities, and system upgrades. Defining a single static baseline that captures the entire lifecycle of an industrial plant is therefore impractical. As a consequence, IDS models trained offline on static datasets rapidly become obsolete when deployed in real environments. To overcome this limitation, we argue that moving beyond traditional offline training paradigms is necessary and that adopting a continual learning approach is essential. Continual learning enables detection models to be incrementally updated using recent, context-specific data, allowing them to adapt to evolving traffic patterns while mitigating catastrophic forgetting. This paradigm is particularly well-suited to IIoT deployments, where long system lifetimes and non-stationary behavior are the norm. In this paper, we propose a cyber threat detection framework that integrates hybrid digital twins (DTs) with continual learning to support adaptive and operationally realistic intrusion detection in IIoT environments. Digital twins serve as local mirrors of industrial devices, keeping sensitive operational data close to the source while enabling controlled validation of security mechanisms before deployment. Continual learning complements this architecture by enabling incremental model updates without repeated full retraining, supporting long-term deployment under bounded computational and memory constraints. The novelty of this work lies in three main aspects: (i) a hybrid digital twin architecture that combines lightweight behavioral abstractions with higher-fidelity replicas for critical assets, enabling realistic yet scalable experimentation; (ii) a continual learning mechanism designed to preserve stable detection performance over time under evolving operational conditions, rather than maximizing offline accuracy; and (iii) an integrated architectural perspective that explicitly addresses adaptability, scalability, and data locality as first-class design objectives for IIoT intrusion detection. The remainder of the paper is organized as follows. Section 2 reviews related work on digital twins and IIoT security. Section 3.1 introduces the threat model considered in this study. Section 4 presents the proposed architecture and methodology. Section 5 describes the experimental evaluation based on benchmark data and a hardware-in-the-loop IIoT use case.

2. State of the art

2.1. Digital twin in Industry 4.0, strengths and applications

The Digital Twin (DT) concept has significantly evolved within the Industry 4.0 context, emerging from manufacturing as an accurate digital representation of physical products throughout their lifecycle. Initially conceptualized for detailed product lifecycle management, DT now supports comprehensive design, prototyping, simulation, production monitoring, and operational optimization [2,3]. DT technology profoundly impacts industrial operations by enabling operational intelligence, predictive maintenance, and productivity optimization. Attaran et al. [4] extensively discusses the role of DT in uncovering operational intelligence within Industry 4.0 scenarios. Further expanding DT applications, Hakiri et al. [5], Bucchiarone [6] provide a comprehensive survey highlighting the transformative role of DTs across industries, from manufacturing to smart cities. They emphasize DTs' ability to enhance operational efficiency, reduce downtime, and enable proactive maintenance strategies through real-time monitoring and predictive analytics.

As industrial environments adopt interconnected devices and real-time communications, they become attractive targets for sophisticated cyberattacks [7]. These attacks can exploit vulnerabilities at different network stack layers and compromise the integrity, confidentiality, or availability of industrial services. IIoT has not been immune to these attacks, as they often rely on protocols that are unsafe by default and difficult to update, which leads to the spread of disruptive attacks [8].

To address these issues, recent research explored several defense strategies. Zolanvari et al. [9] introduced TRUST XAI, a model-agnostic framework for explainable Artificial Intelligence (AI) in IIoT cybersecurity, facilitating interpretability and trust in anomaly detection models. Their solution demonstrated high accuracy and transparency, allowing security operators to understand and validate AI decisions.

Feng et al. [10] emphasized the use of game theory to manage IIoT network vulnerabilities. Their approach models interactions between attackers and defenders, enabling dynamic and proactive defense strategies through patch management and optimal resource allocation. Additionally, Jiang et al. [11] integrates federated learning with blockchain to provide decentralized, tamper-resistant anomaly detection while preserving data privacy. Their architecture effectively balances scalability and security, addressing the specific demands of DT-enabled IIoT infrastructures.

More recently, the integration of AI and Machine Learning (ML) with DT has emerged as a powerful strategy to enhance anomaly detection, predictive analytics, and system optimization within IIoT networks.

In particular, ML techniques have demonstrated substantial effectiveness in anomaly detection and cybersecurity in IIoT environments. Zolanvari et al. [12], Ferrag et al. [13] successfully employed ML techniques to detect cybersecurity real-time attack, significantly enhancing IIoT security infrastructure. Recent literature emphasizes DT-based frameworks that leverage ML for anomaly detection at the network edge, improving real-time responsiveness and accuracy [5,14,29].

Federated learning, a decentralized ML approach, is notably gaining traction in preserving data privacy and efficiently managing distributed IIoT devices [15]. Jiang et al. [11] proposes a sophisticated federated learning framework integrated with a DAG-based blockchain and sharding, thereby improving security, computational efficiency, and scalability. However, modern industrial networks, characterized by dynamic and evolving data streams, require ML models capable of continuous adaptation. Continual Learning (CL), or lifelong learning, has been proven able to address this need, through methods such as Elastic Weight Consolidation (EWC), memory

replay techniques, and dual-memory structures, by enabling incremental knowledge acquisition without forgetting previously learned information [16,17].
Recent studies [17,18] demonstrate the necessity of these techniques for real-time anomaly detection in dynamic IIoT environments, where system behavior continually evolves. Integrating CL with DT technologies further enhances anomaly detection by providing realistic scenarios for incremental learning and continuous system improvement.
These innovations collectively point toward a new generation of intelligent, resilient, and adaptive security mechanisms tailored for complex IIoT ecosystems, grounded in the integration of digital twins, AI, and advanced cybersecurity models.

2.2. Emerging technologies, challenges and future directions

Several emerging technological directions are increasingly shaping the role of DTs within Industry 4.0 and IIoT ecosystems. Blockchain, particularly DAG-based architectures with sharding, is being leveraged to strengthen trust, security, and optimization in DT [11]. Game-theoretic approaches are similarly gaining traction, offering a formal basis for modeling strategic behavior in adversarial environments; for instance, Feng et al. [10] applied evolutionary game theory to characterize vulnerabilities, threat responses, and adaptive patch strategies, thereby improving network resilience. Moreover, edge computing continues to be integrated into DT architectures to support real-time analytics by processing data at or near its source, reducing latency and enabling rapid anomaly detection [5]. Collectively, these technologies reinforce DT systems' capacity to support secure, efficient, and interpretable industrial operations.
Despite these advances, DT-enabled IIoT deployments continue to confront persistent challenges that limit large-scale, reliable adoption. Scalability remains a core concern as industrial systems grow more complex and interconnected. Real-time performance constraints also endure, particularly regarding resource allocation and ultra-low-latency analytics. Interoperability and standardization issues further complicate integration across heterogeneous devices, protocols, and vendor ecosystems. Finally, increasing device density heightens the urgency of robust data security and privacy-preserving mechanisms.
Future research should emphasize tighter integration of these emerging enablers with continual learning paradigms, experiment with hybrid DT-edge-cloud architectures, and pursue rigorous real-world evaluations. Addressing these dimensions holistically will be essential for enhancing the resilience and operational reliability of DT-driven anomaly detection in industrial IoT systems.

2.3. Related work

Research on Digital Twin (DT)-enabled security for Industrial IoT (IIoT) environments has expanded significantly, motivated by the need for real-time situational awareness, resilient anomaly detection, and adaptive defense in cyber-physical systems. Early work primarily leveraged DTs to replicate physical system behavior and support predictive maintenance, but more recent studies increasingly integrate these models into intrusion detection and security monitoring pipelines.
Several studies have explored the use of decentralized or privacy-preserving architectures to enhance trust and scalability in DT-assisted analytics. For example, Jiang et al. [11] proposed a DAG-based blockchain with sharding to support secure federated learning updates in DT environments, addressing issues of distributed trust and data integrity. Complementary to this line of work, Zolanvari et al. [9] introduced a model-agnostic explainer that improves interpretability for IIoT anomaly detection, reflecting increasing interest in explainability as a requirement for DT-enabled decision pipelines.
Real-time performance has also become a central concern. Recent edge-driven DT architectures ([5]) emphasize low-latency data processing and on-device inference to ensure timely anomaly detection. At the same time, other work integrates game-theoretic models to model attacker-defender interactions and optimize dynamic patching strategies [10]. In parallel, several anomaly-detection studies leverage streaming-based or continual-learning methods for handling rapidly evolving IIoT traffic patterns, including lightweight online learning models, memory-efficient incremental updates, and hybrid DL approaches combining autoencoders, GRUs, or temporal-spatial DT representations.
Despite these advances, much of the existing DT-enabled IDS literature exhibits limitations. Many testbeds rely exclusively on simulated environments or static datasets, limiting the validity of the process. Only a few works incorporate hardware-in-the-loop or mixed-reality execution environments capable of capturing realistic network timing, protocol artifacts, and device behavior under stress. Similarly, although continual learning is recognized as crucial for evolving IIoT systems, existing evaluations rarely investigate its data-efficiency constraints or its practicality under constrained memory and bandwidth budgets.
Our work builds on and extends these research directions in three ways. First, unlike prior DT-based IDS studies that rely primarily on virtualized simulation, we implement a hybrid digital twin architecture combining containerized virtual replicas with hardware-in-the-loop components, enabling higher-fidelity observation of IIoT device behavior and more realistic operational feedback. Second, we provide an explicit evaluation of continual-learning data efficiency, demonstrating that models trained on only 9 MB of carefully selected traffic features achieve accuracy comparable to models trained on over 1 GB of raw traffic-an aspect typically overlooked in prior studies. Third, whereas existing works often validate anomaly detection using generic or synthetic attack scenarios, our testbed incorporates realistic industrial attack traces, including malicious register manipulation, reconnaissance activities, and denial-of-service floods, thereby providing more substantial evidence of applicability to industrial security contexts.
These distinctions, highlighted in the comparison Table at 1, position our work as an advancement of DT-enabled IDS research, particularly in terms of architectural realism, efficiency, and security relevance.

Table 1
Comparison of related work and this proposal on DT-enabled IIoT anomaly detection and cybersecurity features.

Work	DT Architecture	Anomaly Detection Method	Real-Time Capability	Realistic Attacks	Key Limitation
[11]	Virtual DT with blockchain-based FL	Federated learning models	Partial (edge-assisted)	No	Focus on trust/scalability rather than IDS performance
[9]	Virtual DT with XAI integration	ML models with model-agnostic explainers	No	Limited	Lacks hardware realism; no continual learning analysis
[5]	Edge-driven DT	Lightweight ML inference at edge	Yes	No	Focus on latency; no hybrid physical-virtual testbed
[10]	DT-informed network model	Game-theoretic vulnerability/patch modeling	Indirect	No	Not an IDS; models strategies rather than detection
Proposed Work	**Hybrid DT with containerized + hardware-in-the-loop components**	**Continual learning for efficient anomaly detection**	**Yes (low-overhead CL)**	**Yes (real industrial attack traces)**	**Addresses gaps in realism, data efficiency, and evaluation depth**

3. Problem statement and threat model

Industrial plants are widely recognized as complex systems composed of heterogeneous technologies. From industrial devices to communication protocols, it is common to find technologies with a time gap of up to 20 years coexisting within the same infrastructure. Industrial systems are known for their long component lifecycle, which has led to several critical challenges, especially with the advent of Industry 4.0. One of the most pressing issues is cybersecurity, specifically the ability of an industrial system to remain resilient and protected against cyber threats targeting its devices and network [19]. Therefore, testing and validating hardware and software solutions for industrial systems is of paramount importance-not only for IT systems in general but particularly for Operational Technology (OT) environments, which are often highly sensitive and high-risk (e.g., power plants) [1,20].

The testing phase in developing and integrating new industrial components is crucial yet challenging. Testing in production environments is risky, particularly for industrial systems. Disrupting business continuity is a significant concern, as is ensuring the safety of the entire plant during testing activities.

For these reasons, testing in digital environments is a highly valuable and necessary approach for industrial systems. However, creating and utilizing a Digital Twin for industrial environments is complex. Not all technologies present in an industrial plant can be easily virtualized (e.g., Programmable Logic Controllers (PLCs)), and network configurations in digital twins often fail to replicate the real system's characteristics fully [21].

Finally, hardware component testing cannot be performed entirely in a digital environment; it requires integration with the real-world industrial scenario using the Hardware-in-the-Loop (HIL) paradigm. This brings us to the central issue: currently, no Digital Twin solutions are specifically designed for testing and security assessment in IIoT environments while integrated within the Cyber Range paradigm. The main challenge is developing an anomaly detection and analysis system capable of integrating traffic generated by the digitalized part of the system with real traffic from the hardware components under test, particularly for an AI-based detection algorithm [22].

3.1. Threat model

We illustrate the system and threat model underlying this work using the Purdue Enterprise Reference Architecture [23] for ICS system security. As depicted in Fig. 1, the Purdue Architecture organizes the ICS network into six layers: levels 4 and 5 form the Information Technology (IT) network. In contrast, the lower levels constitute the Operational Technology (OT) network. The latter handles the control, monitoring, and automation of physical processes. At level 0, sensors and actuators are deployed to interact with the physical process: we refer to them as IIoT devices. They are directly connected to level 1, which comprises various Programmable Logic Controllers (PLCs), an Industrial Gateway aimed at aggregating and exposing Level 0 data, and specific network devices for traffic management, such as Mirroring. These devices implement system control logic by observing sensor readings and updating actuator signals. At the enterprise level (Levels 4–5), threats primarily originate from IT-based attacks, including phishing campaigns, ransomware, and unauthorized data exfiltration. Attackers may compromise enterprise systems to move laterally toward operational technology (OT), exploiting weak segmentation between IT and OT networks. The demilitarized zone (DMZ) at Level 3.5 plays a critical role in isolating these two domains, but misconfigured or vulnerable firewall policies can serve as entry points for adversaries.

Within the operations and control layers (Levels 2–3), the attack surface expands to SCADA systems, human-machine interfaces (HMIs), and programmable logic controllers (PLCs). Here, cyber threats manifest as remote code execution, manipulation of process logic, or injection of rogue commands. Threat actors may exploit protocol weaknesses (e.g., Modbus or specific IoT ones such as MQTT) to interfere with automation processes, leading to process instability or production downtime.

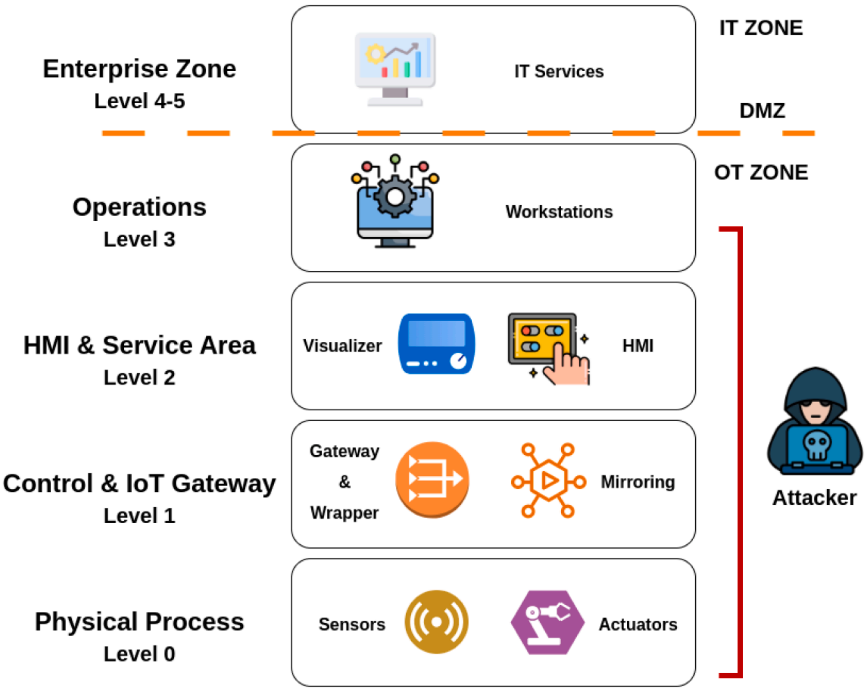

Fig. 1. The threat model of this work is based on the ICS Purdue Reference Architecture. We assume the attacker to have Dolev-Yao capabilities above level 1, to tamper with IIoT traffic data and trigger disruptive and dangerous behavior on the ICS.

At the field device and physical process layers (Levels 0–1), IIoT sensors and actuators become primary targets. Attackers can manipulate sensor readings through spoofing or jamming techniques, leading to erroneous process adjustments or even catastrophic failures in industrial systems. These low-level attacks are particularly challenging to detect because they often blend in with normal operational variations.

To counter these threats, an effective attack detection mechanism must integrate anomaly-based monitoring, AI-driven threat detection, and zero-trust security principles across all Purdue levels. Leveraging real-time analytics and industrial threat intelligence can enhance visibility and resilience against evolving cyber threats.

Our system model assumes network communication between levels 1 and 2 is enabled via an industrial-managed switch. Concerning the attacker model, we argue that the ICS network can be partially controlled by a Dolev-Yao intruder [24]. A typical Dolev-Yao intruder can access all public network messages and modify, inject, delete, or delay them. The Dolev-Yao attacker, therefore, has almost infinite capabilities. However, the intruder is constrained by the perfect cryptography assumption: he can only decrypt a ciphertext or forge a signature if they possesses the corresponding keys. In summary, the attacker cannot carry out cryptographic attacks (e.g., brute force on Private Keys or dictionary attacks on passwords). Additionally, aligning our work with prior work in IIoT security [25–28], we assume the attacker can operate only at or above level 1 of the Purdue Enterprise Reference Architecture.

The attacker's goal is to perform one of the following three attacks:

1. Traffic Manipulation and Tampering: these attacks involve intercepting, modifying, or injecting malicious data into the communication flow, often through Man-in-the-Middle (MITM) attacks. An adversary compromises network traffic by exploiting insecure communication protocols or weak encryption mechanisms, allowing them to alter operational commands or falsify telemetry data. This can lead to incorrect control actions, operational inefficiencies, or even deliberate sabotage, such as forcing a production line to exceed safety thresholds or misleading predictive maintenance algorithms.
2. Sensor disruption attacks exploit vulnerabilities in sensor networks to interfere with data acquisition, either by jamming signals (denial-of-service) or manipulating sensor outputs to generate misleading data. Attackers may use radio frequency (RF) interference to jam wireless sensor communication (e.g., Zigbee, LoRaWAN), spoof environmental readings, or tamper with sensor firmware to falsify operational parameters. This can cause faulty automation decisions, incorrect alarms, production downtime, or even safety hazards (e.g., incorrect temperature readings in chemical processing).
3. Unauthorized actuator manipulation attack: adversaries gain control over industrial actuators to alter physical processes, often through exploitation of insecure remote access or weak authentication. By exploiting vulnerabilities in industrial protocols (e.g., Modbus/TCP, MQTT), default credentials, or misconfigured remote access systems, an attacker can issue unauthorized commands to actuators, disrupting operations. This can lead to dangerous consequences, such as altering robotic arm movements, adjusting industrial conveyor belt speeds, or even manipulating critical infrastructure (e.g., opening valves in a chemical plant or altering turbine speeds in power generation).

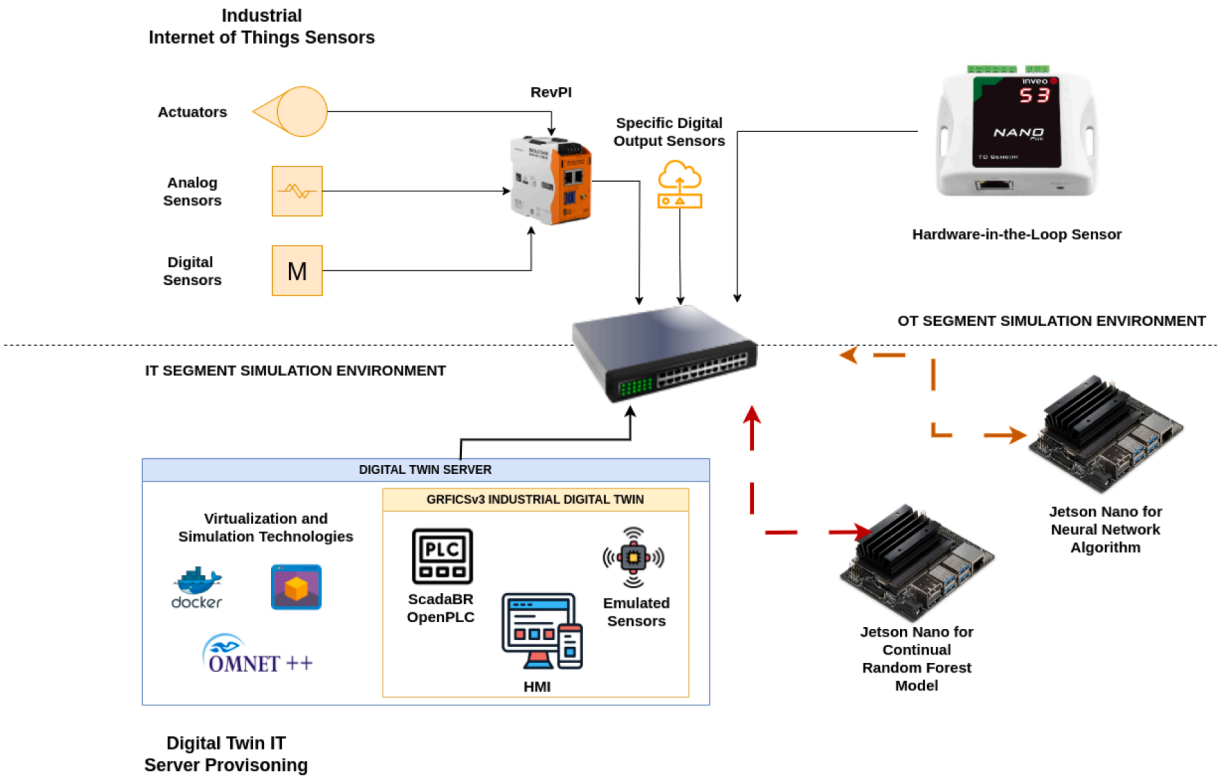

Fig. 2. The digital twin-based IIoT architecture, which follows the classical scheme of the IT segment, where all components, such as PCs, servers, printers, etc., are represented, and the OT segment, which instead represents all components of the industrial network.

4. Architecture and methodology

The architecture we proposed and implemented is shown in Fig. 2. It divides the scenario according to the classical scheme of the IT segment (Information Technology at the bottom), where all components, such as PCs, servers, printers, etc. are represented, from the OT segment (Operational Technology at the top), which instead represents all components of the industrial network, such as sensors, PLCs, industrial gateways, etc.

The IT segment was primarily realized through two key hardware components. In the lower right, we have a Jetson Nano[1] used to run and test the machine learning algorithms once towed. For a more efficient network segmentation of the testbed, we decided to use two Jetson Nano, one for the neural network test and another for the Continual Network Learning tests. The Jetson Nano is connected to the switch with a bi-directional arrow because, in this case, traffic is mirrored from the switch to the Jetson Nano. In the lower left corner, on the other hand, a server is used as a provisioning infrastructure to deploy and run emulation and simulation technologies. The server, as of now, is configured with an OpenStack to manage Virtual Machines and containers, but is intended to be as agnostic as possible from the provisioning platform.

In contrast, the OT segment is divided into two areas. At the top left, the most important one is an industrial RevPi[2] device used to emulate various components of an industrial network, such as a PLC or Edge Node. We chose RevPi due to its open-source and open-hardware architecture and software, enabling us to load industrial firmware and software in a way that simulates its characteristics to the fullest. In addition, the RevPi includes all the necessary gateway modules for digital input and sensors, which we used to connect and test several sensors and generate different industrial protocol traffic, such as Modbus or CAN. Otherwise, the right top section is a network segment reserved on the switch for testing vertical devices that communicate directly via the Ethernet/IP protocol. In this work, we utilized an embedded IoT device capable of using various industrial protocols[3].

The final architecture workflow is the following. Each IIoT device in the monitored environment is associated with a hybrid digital twin that mirrors its operational behavior. The DTs are constructed by continuously collecting device-specific data such as network traffic patterns, sensor readings, and control commands. Instead of replicating every device in full detail, the system differentiates between critical assets, which are modeled with high-fidelity simulation or virtualization (e.g., using containerized replicas of PLCs

[1] https://developer.nvidia.com/embedded/jetson-nano
[2] https://revolutionpi.com/en/revolution-pi-series
[3] https://sklep.inveo.com.pl/en/monitoring/49-nano-temp.html

Table 2
Classification report summarizing model performance.

Class	Precision	Recall	F1-Score	Support
Benign (0)	0.97	1.00	0.98	100,298
Malicious (1)	1.00	0.97	0.98	91,161
Accuracy			0.98	
Macro Avg	0.98	0.98	0.98	191,459
Weighted Avg	0.98	0.98	0.98	191,459

or SCADA services), and non-critical assets, which are represented by lightweight behavioral abstractions. Synchronization between physical devices and their DTs is performed at configurable intervals, ensuring timely reflection of operational states while limiting computational overhead.

4.1. Dataset preparation and model training

Part of the dataset used in this study was obtained from the publicly available IEEE Dataport repository [30]. Specifically, we utilized the Cyber Security Modbus ICS dataset[4], which contains both benign and malicious Modbus network traffic.
This dataset was generated in a small-scale process automation scenario using MODBUS/TCP equipment, designed for research on applying machine learning techniques to cybersecurity in Industrial Control Systems (ICS). The testbed emulates a Cyber-Physical System (CPS) controlled by a SCADA system through the MODBUS/TCP protocol. The setup consists of a liquid pump, simulated by an electric motor whose speed is regulated by a variable frequency drive. The drive, in turn, is controlled by a PLC, which adjusts the motor speed based on predefined liquid temperature thresholds. The temperature measurement is provided by a MODBUS Remote Terminal Unit (RTU), which features a simulated temperature gauge implemented with a potentiometer connected to an Arduino.
The PLC communicates horizontally with the RTU, offering valuable insights into the impact of such communication on the system's overall behavior. Additionally, the PLC interfaces with the Human-Machine Interface (HMI), which oversees and controls the system. This experimental setup provides a realistic representation of industrial automation environments, enabling the collection of network traffic data under both normal and attack scenarios. We first separated the benign traffic from the malicious traffic to preprocess the dataset. Then, we extracted the corresponding CSV files from the original PCAP traces. This created the `merged_traffic.csv` file, which includes only Modbus traffic and volumetric attacks.
Enriched with additional non-volumetric malicious Modbus traffic generated by the described Digital Twin-based IIoT Architecture, we created a second dataset file, `combined_output.csv`, containing a more diverse set of malicious traffic instances. Using these two sources, we built a final dataset composed of:

- 500,000 benign instances,
- 400,000 volumetric malicious instances,
- 100,000 non-volumetric malicious instances.

To ensure an unbiased training process, we performed a random shuffle to mix all instances. This balanced dataset was then used to train the neural network model for intrusion detection. The model training process yielded the following performance results:

- Test Accuracy: 0.9834
- Test AUC: 0.9834
- Precision: 1.0000
- Recall: 0.9653
- F1-Score: 0.9823
- ROC-AUC: 0.9834

The classification report is shown in Table 2, and the corresponding accuracy plots, model loss, and confusion matrix are illustrated in Figs. 3–5, respectively.

4.2. Baseline with continuous learning

In this section, we explain how the detection phase was enriched through continuous learning to train the neural network.
The rationale behind this approach is straightforward. It is insufficient to train a neural network solely on a malicious dataset and to evaluate a new component using predefined, well-known attacks. Such an approach fails to consider several key aspects specific to the IIoT domain. IIoT environments are inherently non-stationary: traffic patterns evolve as production cycles vary, firmware is updated, and new devices or sensors are introduced. A model trained once on historical data rapidly becomes obsolete. In addition, an IIoT system, beyond being vulnerable to particular process or asset-level exploits, is also prone to anomalies tied to the operational

[4] https://ieee-dataport.org/documents/cyber-security-modbus-ics-dataset#files

dynamics of various zones within the industrial facility. For example, attacks on the supply chain may involve subtle manipulation of individual parameters (e.g., a temperature reading or an actuator signal), so that the impact of the intrusion is distributed over time. A single, clearly defined pattern cannot recognize it.

For this reason, it is essential to embed the concept of anomaly detection within the neural network model. This enables the system to identify more sophisticated, long-lasting attacks that may evade traditional pattern-based detection. The neural model is designed to capture temporal and statistical patterns in industrial communication flows, including packet sizes, inter-arrival times, and protocol-specific attributes (e.g., Modbus function codes). The model's output is an anomaly score that distinguishes benign traffic from potential threats. However, to detect anomalies, it is first necessary to define and characterize the plant's baseline behavior-namely, the expected operational profile from which deviations can be identified as anomalous (e.g., unexpected register reads at irregular times or frequencies).

Defining such a baseline is non-trivial. A typical industrial plant exhibits significant variability in production flows, driven by customer orders, seasonal factors, or operational changes. Even year-over-year shifts in company performance-whether in growth or decline-can alter the average behavioral patterns of the infrastructure. Training a neural network on such a vast, non-stationary dataset (e.g., by incorporating 1 year of benign traffic) is highly complex. Moreover, it may not yield reliable or generalizable results.

To address these challenges, we enhanced our architecture by integrating an explicitly dedicated continual learning phase to capture the evolving baseline of the industrial environment. This phase supplements the conventional model training, typically based on static datasets, with continuous, real-world traffic learning, allowing the model to adapt over time to operational changes while retaining the ability to detect anomalies effectively.

Following the approach described in Di Cicco et al. [31], we implement a baseline model called Continual Random Forest (CRF). This approach extends a traditional Random Forest (RF) by incorporating a replay buffer that stores the most informative past examples. For active learning, we also used the approach proposed by the same author, adopting a vote-count-based query strategy and selecting data points for retraining when the model's prediction confidence falls below a 90% majority threshold. The model is retrained after each query, a process that remains computationally efficient due to the lightweight nature of RFs. The CRF is used solely for comparative purposes, allowing us to contrast the performance of a tree-based continual learning method with our neural network model. This ensures methodological consistency and demonstrates that the proposed framework is not tied to a single algorithm but can be generalized across different model families.

4.3. Methodology

The starting point was a neural network model trained on a labeled dataset composed of benign IIoT traffic. We exposed the model to 1 h of real traffic data to evaluate its adaptability under real-world conditions, yielding approximately 1 GB of additional unlabeled data. As a comparative approach, we also conducted a second experiment using only 9 MB of data, extracted by selecting a limited subset of informative features from the full traffic capture. The goal was to assess whether a lightweight representation of traffic data could support continual learning with minimal performance loss. The module consists of three tightly integrated components: (i) a replay buffer storing representative historical samples; (ii) a weight-regularization mechanism to mitigate catastrophic forgetting; and (iii) an adaptive lightweight update strategy.

4.3.1. Replay buffer and sample management

The replay buffer maintains a compact set of previously observed traffic instances without storing full packet traces. Only semantic, protocol-level features (e.g., *Service*, *Conn_state*, *SYN/FIN* flags, Modbus function codes) are retained, significantly reducing storage footprint, as also suggested in Parisi et al. [16]. New samples are inserted when the model's predictive confidence falls below a majority threshold (90%), following a vote-count strategy similar to Hayes and Kanan [17]. This ensures that the buffer captures the most informative boundary cases. During online updates, mini-batches combine new samples with replayed data, stabilizing training by anchoring updates to previously learned patterns.

4.3.2. Weight regularization for knowledge preservation

To mitigate catastrophic forgetting-an inherent issue in continual learning [16]-the module employs an Elastic Weight Consolidation (EWC)-inspired regularization term. After each update, parameters critical to prior tasks are assigned high importance weights (Fisher information). The model was incrementally updated using mini-batches of newly observed traffic in an online learning fashion. Despite the disparity in data volume, both configurations achieved performance comparable to that of the original model. Notably, the model trained on 9 MB of extracted features achieved only a 0.06% reduction in classification accuracy compared to the baseline. Moreover, it exhibited significantly reduced training time and resource usage, shown in Table 3, suggesting that selective feature-based continual learning may offer a viable trade-off for resource-constrained IIoT environments.

Performance metrics-including accuracy, precision, recall, and F1-score-were computed for the original model, the model trained on full traffic, and the model trained only on the selected features. The experimental results, summarized in the following tables and visualizations, demonstrate the robustness of continual learning approaches in IIoT threat detection and highlight the effectiveness of compact data representations in preserving model performance while improving scalability.

To enable continual learning in a real-time Industrial IoT (IIoT) environment, we implemented a lightweight feature-extraction pipeline that processes traffic as it is captured. Instead of feeding raw packets to the model, we derive a set of protocol-aware features that summarize the semantics of each connection. Specifically, we extract the following fields: `Protocol`, `Service`, `Conn_state`, `Is_SYN_ACK`, and `FIN_or_RST`.

Table 3
Continual learning results based on accuracy, precision, recall F1-score, training time and the comparision on the input size.

Model	Accuracy	Precision	Recall	F1-Score	Training Time (s)	Input Size (MB)
Baseline	0.9750	0.960	0.970	0.965	0	500
Continual Learning (1GB)	0.9730	0.958	0.968	0.963	720	1000
Continual Learning (9MB Feat.)	0.9745	0.959	0.969	0.964	45	9

These features are extracted using a custom parser based on Zeek NIDS. The `Protocol` field identifies the transport-layer protocol (e.g., TCP, UDP), while `Service` attempts to label the application-layer service (e.g., Modbus, HTTP). The `Conn_state` field summarizes the connection lifecycle, indicating whether a connection is established, rejected, or incomplete. `Is_SYN_ACK` is a binary feature denoting the presence of a SYN-ACK handshake, typically indicating a successful connection attempt, and `FIN_or_RST` flags whether a session termination has occurred via FIN or RST packets.
These features have been preprocessed (e.g., one-hot encoded or normalized) and fed to the continual learning model in real time, using mini-batches to incrementally update the model's parameters. This streaming setup allows the model to adapt on-the-fly to evolving traffic patterns, enhancing its capability to detect emerging or previously unseen threats with minimal latency and resource overhead.

5. HITL use case test: new sensor

This section presents the use case implemented to validate our proposed framework. The use case aims to perform network fuzzing and stress testing of a new device in a simulated environment using real traffic. The device under test is the sensor shown in Fig. 2, a standard temperature sensor supporting various industrial communication protocols.
The experimental procedure is divided into two main phases. The first phase focuses on assessing the accuracy of the neural network model used for real-time anomaly detection. The second phase aims to evaluate the sensor's behavior when exposed to malicious traffic, specifically analyzing its response when exploited to inject anomalous values into the plant's operation.

5.1. Justification of the binary classification setting

Although multi-class intrusion detection has been widely explored in the literature, this work intentionally adopts a binary classification setting (normal vs. anomalous traffic). This choice is not a limitation of the proposed framework, but a deliberate design decision aligned with the operational requirements of real-world Industrial Intrusion Detection Systems (IDSs).

First, most industrial IDS deployments are detection-oriented rather than attribution-oriented. In safety-critical IIoT environments, the primary objective is to identify anomalous behavior that may indicate a security breach in a timely manner, regardless of the specific attack category. Binary decision outputs enable faster response, simpler integration with mitigation mechanisms, and more reliable alerting under strict real-time constraints.

Second, binary classification is inherently more robust to concept drift, which is a defining characteristic of long-lived industrial systems. Attack taxonomies evolve over time, new threat classes emerge, and previously known attacks may change their signatures. Under these conditions, maintaining a stable multi-class labeling scheme becomes impractical. Conversely, the normal-versus-anomalous boundary remains comparatively stable, making binary detection better suited for continual learning scenarios.

Finally, the binary setting significantly reduces reliance on continuous, fine-grained labeling, which is particularly costly in industrial contexts. Accurate multi-class annotations require domain expertise and frequent dataset updates, whereas binary labels can be obtained with less supervision and are better suited to incremental learning driven by Digital Twin-generated data.

For these reasons, the adopted binary classification setting enhances the scalability, adaptability, and deployability of the proposed DT-enabled continual learning framework, without restricting its effectiveness in practical IIoT security applications.

5.2. Neural network validation

The neural network was trained on the preprocessed dataset, which was first balanced using the Borderline-SMOTE technique to address class imbalance. The model architecture consisted of multiple dense layers with ReLU activation functions, interleaved with batch normalization layers to stabilize training and dropout layers for regularization. Specifically, the first thick layer included 128 neurons, followed by a batch normalization layer and a dropout layer with a 30% rate. This was followed by a 64-neuron dense layer with the same normalization and dropout strategy, and finally, a 32-neuron dense layer before the output layer.

The merged dataset was preprocessed to extract the relevant features:

- `Protocol`
- `Service`
- `Conn_state`
- `Is_SYN_ACK`

Table 4
Performance metrics.

Metric	Value
Accuracy	98.34%
Test AUC	98.34%
Precision	100.00%
Recall	96.53%
F1-score	98.23%
ROC-AUC	98.34%

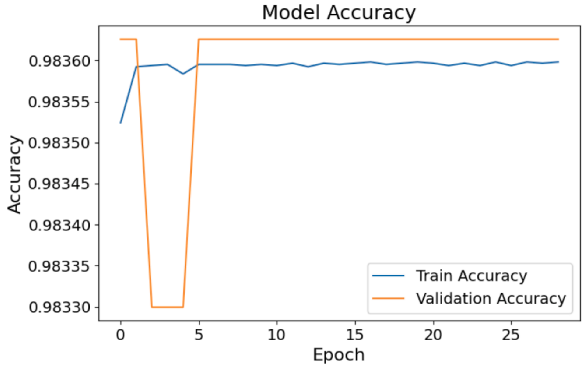

Fig. 3. Model accuracy of the proposed neural network.

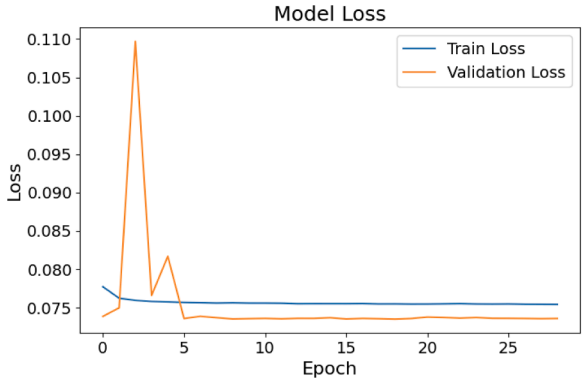

Fig. 4. Model Loss of the proposed Neural Network.

- `FIN_or_RST`
- `anomaly_alert`
- `OSSEC_alert`
- `Login_attempt`
- `read_write_physical.process`

We employed the Adam optimizer with an exponentially decaying learning rate schedule to ensure adequate training. The model was trained using binary cross-entropy loss, with accuracy and AUC (Area Under the Curve) as evaluation metrics. Additionally, class weights were computed and incorporated to further mitigate class imbalance. The final layer was a single neuron with a sigmoid activation function, enabling binary classification of IIoT traffic into normal or malicious categories. The model was trained for 30 epochs, achieving a test accuracy of approximately 98.34%. Table 4 presents the detailed performance metrics, including precision, recall, and F1-score.

Figs. 3 and 4 illustrate the training and validation accuracy and loss over the epochs, respectively.

The trained model demonstrated excellent performance on the test dataset, achieving an overall accuracy of 98.34%. This means that nearly all instances were correctly classified. We analyzed key performance metrics to further assess the model's effectiveness, including precision, recall, F1-score, and the area under the ROC curve (AUC). The precision, which measures the proportion of correctly identified positive instances among all predicted positive instances, reached 100%. This indicates that the model produced no false positives. The recall, which represents the proportion of actual positive instances correctly identified, was 96.53%, suggesting a very low false-negative rate. The F1-score, a harmonic mean of precision and recall, was 98.23%, indicating well-balanced perfor-

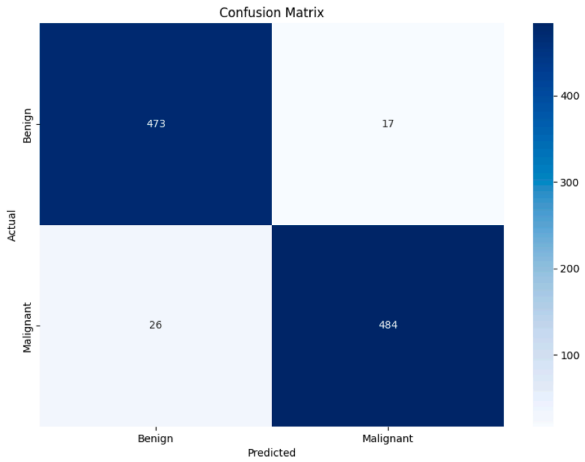

Fig. 5. Class distribution of original and predicted classes.

Table 5
Evaluation metrics for each attack type and learning method.

Attack	Method	Accuracy	Precision	Recall	F1-score
Malicious Write	Full Training	0.987	0.981	0.976	0.978
Malicious Write	Continual Learning	0.985	0.978	0.974	0.976
Register Scan	Full Training	0.973	0.965	0.960	0.962
Register Scan	Continual Learning	0.969	0.959	0.955	0.957
DoS	Full Training	0.951	0.948	0.939	0.943
DoS	Continual Learning	0.947	0.945	0.935	0.940

mance across these two metrics.
The ROC-AUC score, a key indicator of the model's ability to distinguish between benign and malicious traffic, was also 98.34%. This confirms the classifier's robustness in differentiating between the two classes. Breaking down performance by class, the model achieved 97% precision and 100% recall, correctly classifying nearly all regular traffic without mislabeling any malicious instances as benign. For the malicious class, the model achieved perfect precision (100%) and recall of 97%, correctly identifying the vast majority of attacks while missing only a small fraction. The model's strong performance is further illustrated by the classification distribution plot in Fig. 5, which shows the agreement between predicted and actual labels. These results confirm the model's effectiveness in detecting and classifying threats in IIoT environments.

5.3. Experimental evaluation with hardware-in-the-loop modbus sensor

To validate the robustness and adaptability of our anomaly detection system in real-world scenarios, we designed a comprehensive set of experiments involving a newly acquired Modbus sensor connected via hardware-in-the-loop (HIL) to the proposed testbed. The sensor is depicted in the top-right corner of Fig. 2. Each experiment family consists of three attack scenarios emulated directly from the sensor to replicate realistic malicious behaviors:

Replay Attacks. Replay-based manipulations are among the most common threats in industrial networks relying on legacy protocols such as Modbus/TCP, which lack native freshness guarantees. In our testbed, an adversary re-injects previously recorded legitimate frames-e.g., valid temperature readings or actuator state reports-to mislead process controllers.

Command-Injection Attacks. In this scenario, the attacker injects forged Modbus `WRITE_REGISTER` or `FORCE_SINGLE_COIL` commands to manipulate actuator states or override control logic. Unlike raw flooding, these attacks mimic legitimate command patterns but alter critical fields (e.g., function codes, register offsets).

Multi-Point Scanning Attacks. We introduce a coordinated scanning pattern that enables the adversary to probe multiple register ranges and function-code families in parallel. This simulates reconnaissance by compromised gateways or lateral-moving adversaries.

On the sensor, this attack family has been implemented through a set of specific actions:

- **Malicious Register Write Attack:** The sensor sends unsolicited `WRITE_REGISTER` Modbus commands, injecting invalid or harmful values into critical registers.
- **Broadcast Register Scanning Attack:** The sensor performs repeated `READ_HOLDING_REGISTERS` requests targeting a vast address space to probe system memory, simulating reconnaissance.

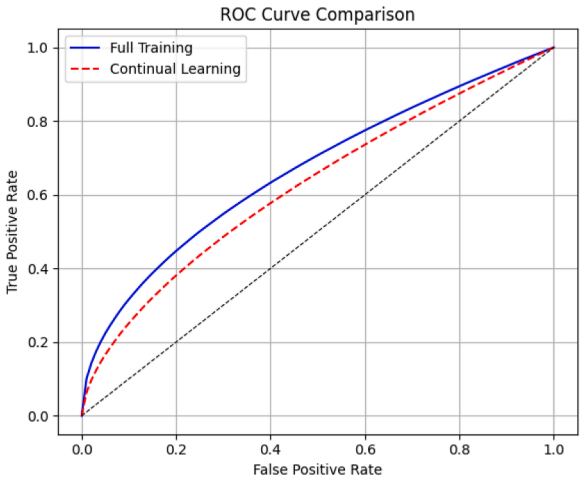

Fig. 6. ROC curve comparison between full and continual versions.

- **Denial of Service (DoS) Attack:** A flood of Modbus requests is generated to saturate the network or overload the controller, emulating resource exhaustion.

The attack and traffic simulation have been performed using the following tools:
`Modbussploit`[5] was employed for the generation of attacks, specifically using the following modules:

- **ReadSingleRegister** in broadcast mode,
- **WriteSingleRegister** for writing to a single register,
- **DoSCoil** for performing Denial-of-Service (DoS) attacks.

The `GRFICSv3`[6] chemical plant simulator was used for the plant simulation. Once deployed, it instantiates the following virtual machines:

- A VM running `OpenPLC`[7], responsible for managing the logic controller part of the plant,
- A VM running `ScadaBR`[8], responsible for automation and supervision,
- A VM that virtualizes the actuators and sensors of the plant's pumps and valves.

By modifying the third VM, we integrated a **physical hardware sensor** into the digitized chemical plant, enabling hardware-in-the-loop testing.
We tested each attack under two distinct learning paradigms:

1. **Standard Full-Dataset Training:** The neural network is pre-trained on the full labeled dataset, including synthetic traffic generated by the digital twin. No incremental updates are allowed during testing.
2. **Continual Learning:** The neural network is initialized from the baseline model and incrementally updated using a stream of selected traffic features. Only a small portion of the data is used to simulate realistic deployment constraints.

Both setups are evaluated using identical traffic inputs over 1 day of plant uptime (24 hours), combining digital twin and live HITL sensor data. The metrics recorded include detection accuracy, false positive rate, and training data volume.

The overall results are shown in Table 5. They show, for each threat attack, the accuracy, precision, and recall of both models: one trained with classic training on all the traffic and the other with the Continual Learning Approach. The results reveal that the continual learning approach achieves accuracy comparable to that of the full-training model across all attack scenarios. In contrast, the full-training model performs better only for volumetric attacks, where detection time is less deterministic. For the same reasons, the ROC Curve comparison in Fig. 6 of the two models does not show particular differences in false positive rate.

However, the real advantage is shown in Fig. 7, where the continual approach requires approximately 20 times less data to reach optimal performance, demonstrating its efficiency and applicability in data-scarce industrial environments. The graph, in particular, must be read as follows: the blue straight line represents the data collected over 1 day (24 hours), which is used as a single block to train the neural network model, approximately 20 GB. The red, dashed line represents the data used to feed the model with

[5] https://github.com/C4l1b4n/ModBusSploit
[6] https://github.com/mrideout/GRFICSv3
[7] https://autonomylogic.com/
[8] https://github.com/ScadaBR

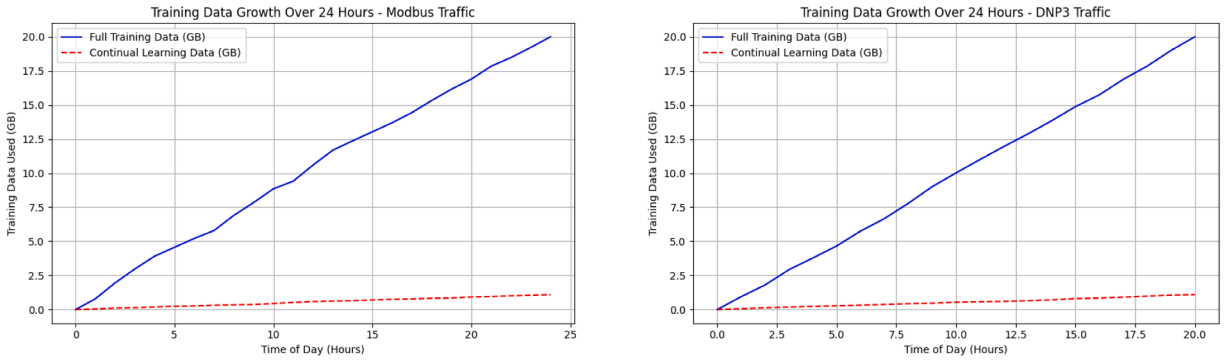

Fig. 7. Training data used: comparison between full and continual versions for Modbus Traffic on the left and DNP3 traffic on the right.

Table 6
Ablation study comparing DT ON and DT OFF configurations.

Configuration	Accuracy	F1-score	FPR
DT OFF (CL only)	0.9684	0.9673	0.0240
DT ON (DT + CL)	0.9240	0.9199	0.0505

the continual learning approach. In 24 hours, it will be trained and have less than 2 GB of data. For additional clarification and to strengthen the validity of our findings, we replicated the CL experiments using DNP3 traffic instead of Modbus by enabling the dedicated DNP3 sensor in the architecture and generating protocol-compliant activity through a Python script[9]. This secondary evaluation demonstrates that the behavior of our CL pipeline is not protocol-dependent: the model continued to perform stable incremental updates, required only a minimal proportion of the available traffic, and preserved its efficiency even under a markedly different industrial communication standard. Notably, the feature-selection strategy retained its discriminative power despite the shift from Modbus's request-response semantics to DNP3's event-driven characteristics, indicating robust protocol-agnostic representation learning. The resulting performance closely mirrored that obtained in the Modbus setting, and is shown on the right part of Fig. 7. Altogether, the DNP3 experiments confirm that the percentage of traffic required for effective CL remains similarly low, further supporting the practicality of our approach in resource-constrained industrial deployments.

5.4. Ablation study: impact of the digital twin

To explicitly quantify the contribution of the Digital Twin (DT) component within the proposed framework, we conduct a controlled ablation study comparing two configurations:

- **DT ON**: the complete framework, where continual learning is driven by traffic generated through the hybrid Digital Twin, including virtualized components and hardware-in-the-loop devices;
- **DT OFF**: a reduced configuration where the same continual learning pipeline operates on a static, offline dataset without DT-mediated traffic generation or synchronization.

It is important to note that the DT ON/DT OFF distinction applies exclusively to the data generation and synchronization mechanism. Both configurations employ the same continual learning strategy with replay and regularization, unless explicitly stated otherwise in the catastrophic forgetting analysis. Both configurations employ the same model architecture, feature set, replay buffer capacity, and training hyperparameters. This ensures that any observed differences can be attributed solely to the presence or absence of the Digital Twin component.

5.4.1. Experimental protocol

The dataset is partitioned into four sequential tasks in temporal order, emulating the progressive operational phases of an IIoT plant. After training on each task, the model is evaluated on all previously observed tasks to assess both detection performance and resistance to catastrophic forgetting. Performance is reported in terms of accuracy, F1-score, and false positive rate (FPR).

5.4.2. Quantitative results

Table 6 summarizes the average performance across all tasks after the final continual learning update.

[9] https://github.com/hpcn-uam/DNP3-Attack-Detection-System

Table 7
Accuracy comparison across incremental updates with and without replay and regularization.

Update	CL with replay & regularization	CL without mitigation	Difference
U1	0.9567	0.9613	-0.0046
U2	0.9112	0.9679	-0.0567
U3	0.9165	0.9667	-0.0502
U4	0.8884	0.9653	-0.0769

This result should not be interpreted as a limitation of the Digital Twin, but rather as evidence of the trade-off between offline optimality and online deployability: DT OFF benefits from unrealistically stable data distributions, while DT ON operates under realistic, evolving, and resource-constrained conditions.

At first glance, the DT OFF configuration achieves higher absolute classification performance. This behavior is expected, as offline training benefits from a stable data distribution and complete historical visibility. However, such conditions are rarely attainable in real industrial deployments.

While DT OFF yields higher accuracy, it cannot safely incorporate evolving operational behavior, firmware changes, or newly introduced devices without retraining on static datasets. In contrast, the DT ON configuration reflects realistic deployment constraints: traffic is generated incrementally, operational baselines evolve over time, and only bounded memory is available for replay.

Notably, despite these challenges, the DT-enabled framework maintains stable performance across successive tasks, with limited degradation on earlier tasks (approximately 3.2% forgetting). This demonstrates that the Digital Twin acts primarily as an *operational enabler* for continual learning rather than as a direct performance booster.

In practical IIoT settings, the DT facilitates: (i) safe generation of realistic traffic patterns without impacting live systems; (ii) controlled exposure to novel behaviors and attack strategies; and (iii) incremental model adaptation under bounded computational and storage constraints.

Therefore, the results of this ablation study highlight a fundamental trade-off: while offline learning maximizes raw detection accuracy, integrating Digital Twins enables the deployment of adaptive, long-lived intrusion detection systems suitable for real industrial environments.

5.5. Discussion

Although the proposed Digital Twin-enhanced continual learning framework demonstrates strong adaptability to evolving IIoT traffic patterns, some limitations remain and motivate future research.

Catastrophic Forgetting. The combination of replay-buffer sampling and regularization mitigates forgetting across incremental updates, independently of the Digital Twin component. Table 7 reports detection accuracy across successive updates (U1-U4) for continual learning with and without forgetting-mitigation mechanisms.

The accuracy trend across updates is visualized in Fig. 8, illustrating the improved stability offered by the proposed approach:

The results confirm that the proposed strategy reduces forgetting and maintains more consistent performance compared to the unmitigated baseline. However, long-term stability under rapid or adversarial drift remains an open challenge. Future work will extend the consolidation mechanism with adaptive importance weighting and hybrid memory strategies to improve robustness under persistent distribution shift.

Lightweight Baseline Comparison. To contextualize our continual learning approach, we compare it against two lightweight intrusion detection baselines suitable for edge deployment: *Random Forest* (RF) and *Isolation Forest* (IF). Both models are trained once on the initial data distribution and evaluated statically on the same incremental test sets. As shown in Table 8, RF achieves consistently high accuracy (∼0.97) across all updates, confirming its strong generalization on static IIoT traffic. However, RF cannot adapt to distribution shifts, making it unsuitable for dynamic threat environments. Conversely, IF performs poorly (accuracy ∼0.52), as expected for an unsupervised anomaly detector operating without labeled updates in a concept-drift scenario. Our proposed continual learning method with replay (CL_with_Replay) maintains competitive accuracy while enabling online adaptation, achieving stability close to RF ($\Delta < 0.07$ after U4) and outperforming IF by a large margin ($\Delta \approx 0.45$). Notably, CL without mitigation (CL_no_Mitigation) shows higher but unsustainable accuracy, as it catastrophically forgets previous knowledge. These results highlight that our approach effectively balances *adaptability* and *stability*, a critical requirement for practical IIoT security systems operating under evolving threats. These baselines are not intended to outperform the proposed method, but to provide a realistic lower-bound reference for edge-compatible IDS solutions.

5.6. Scalability, privacy, and adaptability considerations

Scalability, privacy preservation, and adaptability are key design goals of the proposed DT-enabled continual learning framework. In this work, these properties are addressed at the architectural level rather than being claimed as empirically validated performance metrics.

Scalability. The framework is scalable by design through the use of bounded computational mechanisms. Continual learning updates operate on small mini-batches and a fixed-size replay buffer, ensuring that memory usage and computational overhead

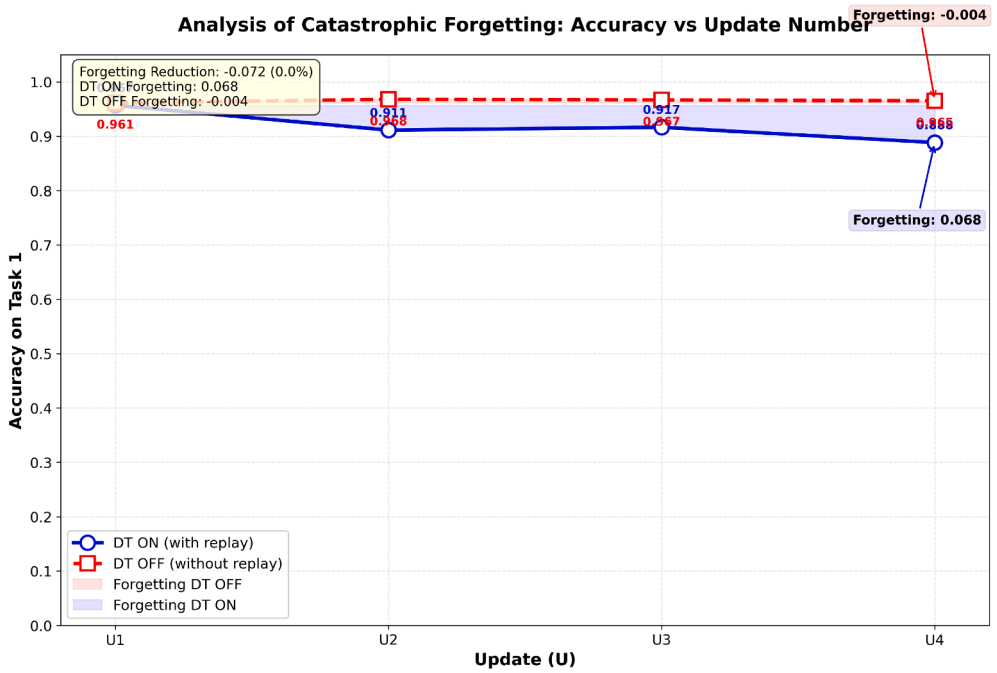

Fig. 8. Accuracy evolution across updates U1-U4 for CL with replay/regularization (DT ON) and without mitigation (DT OFF).

Table 8
Performance comparison between baseline lightweight models and continual learning approaches.

Update	Random Forest	Isolation Forest	CL_with_Replay	CL_no_Mitigation
U1	0.9693	0.5206	0.9567	0.9613
U2	0.9697	0.5221	0.9112	0.9679
U3	0.9698	0.5216	0.9165	0.9667
U4	0.9709	0.5226	0.8884	0.9653

remain bounded as the system evolves. Moreover, the hybrid Digital Twin architecture enables the orchestration of multiple DT instances (e.g., per device, subsystem, or production line), allowing horizontal scaling in large IIoT deployments. A quantitative evaluation of scalability as the number of devices and concurrent DT instances increases is left for future work.

Privacy Preservation. Privacy is addressed through architectural choices rather than explicit privacy-enhancing mechanisms. Raw operational data are retained locally at the Digital Twin or device level, while model updates rely on incremental learning without centralized aggregation of sensitive traffic traces. This design minimizes data exposure and reduces the need for cross-domain data sharing. Formal privacy guarantees and comparisons with federated or encrypted learning schemes are beyond the scope of this study and will be explored in future extensions.

Adaptability. Adaptability is achieved through continual learning, which enables the IDS to incrementally incorporate new traffic patterns and attack behaviors without retraining from scratch. The Digital Twin serves as an operational enabler, providing controlled, evolving data streams that reflect realistic system changes. While the proposed approach demonstrates resilience to distribution shifts, adaptability is currently driven by predefined update schedules and replay strategies. Future work will investigate adaptive update triggering based on explicit drift-detection mechanisms and Digital Twin feedback loops.

Overall, these characteristics should be interpreted as architectural enablers for long-lived and deployable IIoT intrusion detection systems, rather than as fully quantified system-level guarantees.

6. Conclusion

Despite its numerous benefits, the connectivity of IoT devices in industrial environments (IIoT) increases security risks. It significantly expands the threat surface, including DDoS, device hijacking or spoofing, and man-in-the-middle attacks. Therefore, designing and implementing robust security measures is essential to repel cybersecurity attacks. However, testing these strategies is not always possible in a real-world environment due to the sensitivity of the industrial scenarios. For this reason, architecture based on digital twins for Industrial IoT scenarios is paramount for designing, testing, and validating industrial protocols, devices, and effective intrusion-detection mechanisms. In this paper, we presented an IIoT architecture based on the Digital Twin paradigm for testing the

security devices and protocols in the industrial domain.
We developed a neural network trained and tested on this architecture to validate the proposal for detecting malicious traffic from IIoT devices. Here comes another essential contribution to this paper. We demonstrated that the complexity and technological heterogeneity of the plant in the IIoT domain make it challenging to train a model that defines the baseline. We then showcased how a continual learning approach can still achieve great performance, with accuracy of nearly 97%, while using 20x less data than the baseline. These results confirm that the proposed approach effectively detects cyber threats in IIoT environments.
The present evaluation focuses primarily on Modbus/TCP traffic. To assess generalization across heterogeneous IIoT ecosystems, future work will focus on refining the model, testing it on additional datasets, including OPC-UA and MQTT, as well as on multimodal datasets that combine network telemetry with device-side process variables. These research directions will enhance the resilience of the continual learning subsystem and support broader applicability across diverse industrial environments.

CRediT authorship contribution statement

Andrea Melis: Writing – review & editing, Writing – original draft, Software, Formal analysis, Conceptualization; **Andrea Piroddi:** Writing – original draft, Investigation, Formal analysis, Data curation; **Chan-Tong Lam:** Writing – review & editing, Funding acquisition; **Giovanni Pau:** Writing – review & editing, Resources, Project administration; **Roberto Girau:** Writing – review & editing, Writing – original draft, Project administration.

Data availability

Used data for public repository which has been described and referenced in the paper correctly

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported in part by Macao Science and Technology Development Fund (FDCT) under Grant 0033/2023/RIA1

References

[1] D. Berardi, F. Callegati, A. Giovine, A. Melis, M. Prandini, L. Rinieri, When operation technology meets information technology: challenges and opportunities, Future Inter. 15 (3) (2023). https://www.mdpi.com/1999-5903/15/3/95. https://doi.org/10.3390/fi15030095
[2] M. Grieves, Digital twin: manufacturing excellence through virtual factory replication (2015).
[3] M. Grieves, J. Vickers, Digital Twin: Mitigating Unpredictable, Undesirable Emergent Behavior in Complex Systems, 2017, pp. 85–113. https://doi.org/10.1007/978-3-319-38756-7_4
[4] S. Attaran, M. Attaran, B.G. Celik, Digital twins and Industrial Internet of Things: uncovering operational intelligence in Industry 4.0, Deci. Analyt. J. 10 (2024) 100398.
[5] A. Hakiri, A. Gokhale, S.B. Yahia, N. Mellouli, A comprehensive survey on digital twin for future networks and emerging Internet of Things industry, Comput. Netw. (2024) 110350.
[6] A. Bucchiarone, Gamification and virtual reality for digital twins learning and training: architecture and challenges, Virt. Real. Intell. Hard. 4 (6) (2022) 471–486. https://doi.org/10.1016/j.vrih.2022.08.001
[7] A. Aldhaheri, et al., Deep learning for cyber threat detection in IoT networks: a review, IoT Cyber-Phys. Syst. 4 (2024) 100–116. https://doi.org/10.1016/j.iotcps.2023.12.000
[8] M. Benmalek, Ransomware on cyber-physical systems: taxonomies, case studies, security gaps, and open challenges, IoT Cyber-Phys. Syst. 4 (2024) 186–202. https://doi.org/10.1016/j.iotcps.2024.02.002
[9] M. Zolanvari, Z. Yang, K. Khan, R. Jain, N. Meskin, TRUST XAI: Model-agnostic explanations for AI with a case study on IIoT security, IEEE Internet Thing. J. 10 (4) (2021) 2967–2978.
[10] H. Feng, D. Chen, H. Lv, Z. Lv, Game theory in network security for digital twins in industry, Digit. Commun. Netw. 10 (4) (2024) 1068–1078.
[11] L. Jiang, Y. Liu, H. Tian, L. Tang, S. Xie, Resource-efficient federated learning and DAG blockchain with sharding in digital-twin-driven industrial IoT, IEEE Internet Thing. J. 11 (10) (2024) 17113–17127.
[12] M. Zolanvari, M.A. Teixeira, L. Gupta, K.M. Khan, R. Jain, Machine learning based network vulnerability analysis of Industrial Internet of Things, IEEE Internet Thing. J. 6 (4) (2019) 6822–6834.
[13] M.A. Ferrag, F. Alwahedi, A. Battah, B. Cherif, A. Mechri, N. Tihanyi, T. Bisztray, M. Debbah, Generative AI in cybersecurity: a comprehensive review of LLM applications and vulnerabilities, IoT Cyber-Phys. Syst. 5 (2025) 1–20. https://doi.org/10.1016/j.iotcps.2025.01.001
[14] F. Alwahedi, A. Aldhaheri, M.A. Ferrag, A. Battah, N. Tihanyi, T. Bisztray, M. Debbah, Machine learning techniques for IoT security: current research and future vision with generative AI and large language models, IoT Cyber-Phys. Syst. 4 (2024) 167–185. https://doi.org/10.1016/j.iotcps.2023.12.001
[15] A. Gerodimos, L. Maglaras, M.A. Ferrag, N. Ayres, I. Kantzavelou, IoT: communication protocols and security threats, IoT Cyber-Phys. Syst. 3 (2023) 1–13. https://doi.org/10.1016/j.iotcps.2022.12.003
[16] G.I. Parisi, R. Kemker, J.L. Part, C. Kanan, S. Wermter, Continual lifelong learning with neural networks: a review, Neural Netw. 113 (2019) 54–71.
[17] T.L. Hayes, C. Kanan, Lifelong machine learning with deep streaming linear discriminant analysis, Pattern Recognit. 101 (2020) 107200.
[18] H. Zhou, C. Xu, X. Liang, X. Luo, Continual learning for robust deep intrusion detection in dynamic IoT environments, Comput. Netw. 212 (2022) 109004.
[19] M. Serror, S. Hack, M. Henze, M. Schuba, K. Wehrle, Challenges and opportunities in securing the Industrial Internet of Things, IEEE Trans. Ind. Inf. 17 (5) (2021) 2985–2996. https://doi.org/10.1109/TII.2020.3023507
[20] F. Kara, H. Kaya, H. Yanikomeroglu, B.K. Ng, C.-T. Lam, Bit-interleaved multiple access: improved fairness, reliability, and latency for massive IoT networks, IEEE Internet Thing. J. 10 (18) (2023) 16006–16027.
[21] L.P.I. Ledwaba, G.P. Hancke, Security Challenges for Industrial IoT, Springer International Publishing, Cham, 2021, pp. 193–206. https://doi.org/10.1007/978-3-030-51473-0_10. https://doi.org/10.1007/978-3-030-51473-0_10

[22] C. Ni, S.C. Li, Machine learning enabled industrial IoT security: challenges, trends and solutions, J. Indust. Inform. Integr. 38 (2024) 100549. https://www.sciencedirect.com/science/article/pii/S2452414X2300122X. https://doi.org/10.1016/j.jii.2023.100549
[23] T.J. Williams, The purdue enterprise reference architecture, Comput. Ind. 24 (2–3) (1994) 141–158.
[24] D. Dolev, A. Yao, On the security of public key protocols, IEEE Trans. Inf. Theory 29 (2) (1983) 198–208.
[25] S.G. Abbas, M.O. Ozmen, A. Alsaheel, A. Khan, Z.B. Celik, D. Xu, {SAIN}: Improving {ICS} attack detection sensitivity via {State-Aware} invariants, in: 33rd USENIX Security Symposium (USENIX Security 24), 2024, pp. 6597–6613.
[26] M. Ike, K. Phan, K. Sadoski, R. Valme, W. Lee, Scaphy: detecting modern ics attacks by correlating behaviors in scada and physical, in: 2023 IEEE Symposium on Security and Privacy (SP), IEEE, 2023, pp. 20–37.
[27] R. Pickren, T. Shekari, S. Zonouz, R. Beyah, Compromising industrial processes using web-based programmable logic controller malware, in: Network and Distributed System Security (NDSS) Symposium, 2024.
[28] A. Erba, N.O. Tippenhauer, Assessing model-free anomaly detection in industrial control systems against generic concealment attacks, in: Proceedings of the 38th Annual Computer Security Applications Conference, 2022, pp. 412–426.
[29] A. Melis, S. Layeghy, D. Berardi, M. Portmann, M. Prandini, F. Callegati, P-SCOR: Integration of Constraint Programming Orchestration and Programmable Data Plane, IEEE Trans. Netw. Serv. Manag. 18 (1) (2021) 402–414.
[30] I. Frazão, P. Abreu, T. Cruz, H. Araújo, P. Simões, Cyber-security Modbus ICS dataset, 2019. https://dx.doi.org/10.21227/pjff-1a03. https://doi.org/10.21227/pjff-1a03
[31] N. Di Cicco, A. Al Sadi, C. Grasselli, A. Melis, G. Antichi, M. Tornatore, Poster: continual network learning, in: Proceedings of the ACM SIGCOMM 2023 Conference, ACM SIGCOMM '23, Association for Computing Machinery, New York, NY, USA, 2023, p. 1096–1098. https://doi.org/10.1145/3603269.3610855. https://doi.org/10.1145/3603269.3610855