Enhanced Anomaly Detection in Industrial Control Systems aided by Machine Learning ⋆

Vegard Berge, Chunlei Li [ID]
Emails: vegard.berge8@gmail.com, chunlei.li@uib.no

Department of Informatics, University of Bergen, Bergen, 5020, Norway

Abstract. Traditional intrusion detection systems (IDSs) often rely on either network traffic or process data, but this single-source approach may miss complex attack patterns that span multiple layers within industrial control systems (ICSs) or persistent threats that target different layers of operational technology (OT) systems. This study investigates whether combining both network and process data can improve attack detection in ICSs environments. Leveraging the Secure Water Treatment (SWaT) dataset, we evaluate various machine learning models on individual and combined data sources. Our findings suggest that integrating network traffic with operational process data can enhance detection capabilities, evidenced by improved recall rates for cyber attack classification. Serving as a proof-of-concept within a limited testing environment, this research explores the feasibility of advancing intrusion detection through a multi-source data approach in ICSs. Although the results are promising, they are preliminary and highlight the need for further studies across diverse datasets and refined methodologies.

1 Introduction

As industrial environments grow increasingly interconnected, with the proliferation of technologies such as the industrial Internet of Things (IIoT) and Industry 4.0, the security of ICSs becomes ever more critical. ICSs are the backbone of critical infrastructure, overseeing and automating processes in sectors such as energy, water treatment, manufacturing, and transportation. The convergence of information technology (IT) and operational technology (OT) in these systems has improved efficiency but also exposed ICSs to a wide range of cyber threats [4,28]. Cyber attacks targeting ICSs can have catastrophic consequences, ranging from operational disruption to physical damage and even loss of life, as many ICSs components are responsible for real-time control of physical processes. Most conventional IDSs focus on IT environments, emphasising data confidentiality. However, in the OT context of ICS, availability is the paramount concern, as systems must remain continuously operational to avoid disruption of critical

---

⋆ The full version of the work can be found at https://bora.uib.no/bora-xmlui/handle/11250/3140895

services [9]. This shift in priorities, coupled with the unique operational characteristics of ICS, necessitates the development of specialised IDSs tailored to detect both network and process anomalies.

1.1 Research Motivation and Problems

The integration of network traffic data with operational process data is a promising approach to improving the detection capabilities of IDSs in ICS environments. While network-based IDSs (NIDSs) monitor network traffic for suspicious patterns, they often miss anomalies within the physical processes themselves. Conversely, process-based IDSs focus on deviations in the behaviour of the physical components, such as programmable logic controllers (PLCs) and sensors, but may lack the broader context provided by network data. This study explores the potential for enhanced detection capabilities by incorporating both OT and IT data. This preliminary approach is not intended to replace existing detection frameworks but rather to assess the viability of enriching them with complementary data sources. Specifically, this research aims to answer the following questions:

- How can machine learning models be applied to detect anomalies across both network and process data in ICSs?
- Does the integration of these two data types improve the overall detection capabilities and reduce false positives compared to models trained on network or process data alone?
- What are the limitations of current intrusion detection approaches, and how can this research contribute to overcoming them?

1.2 Main Contributions

This research tries to contribute to the field of ICS security in these aspects:

- Integrated Anomaly Detection: We propose an approach that combines network traffic data and process data to improve the detection of cyber attacks in ICS environments. This hybrid approach enables a more comprehensive detection strategy by capturing both network anomalies and operational deviations.
- Evaluation on Real-World Data: Using the Secure Water Treatment (SWaT) testbed dataset from iTrust Labs, this study evaluates machine learning models on realistic ICS data. The SWaT dataset includes both network and process data, providing a rich environment for testing IDSs approaches.
- Machine Learning Models for ICSs: A range of machine learning models, including Decision Trees, Random Forests, Support Vector Machines (SVM), and Neural Networks are applied. This research explores how these models perform binary classification, trained on a combined dataset compared to when they are trained on either network or process data alone.

– Actionable Insights: By evaluating the results from our models, on the variation of datasets, we try to explain what benefits might come from viewing all assets in an ICS as potential *sensor* or *agent* for anomaly detection.

The remainder of this paper is organised as follows: In Section 2, we start with an overview of ICSs, discuss their key components and security challenges, then review related work in the field of IDSs for ICSs. In Section 3 we introduce the SWaT testbed and describe its relevance to the research. Section 4 outlines the methodology used in this study, including data preprocessing and model training. Section 5 presents the experimental results and discusses their implications. Finally, Section 6 concludes the paper and suggests directions for future research.

2 Preliminaries

2.1 Overview of ICSs

ICSs integrate hardware, software, and network systems to monitor, control, and automate physical processes. These systems are essential for managing large-scale industrial operations that require precise and continuous control, where downtime can have severe economic, environmental or safety impacts [28].

ICSs consist of several core components that interact to ensure efficient, reliable, and safe industrial operations. Some of these components include:

– **Programmable Logic Controllers (PLCs)**: programmable logic controllers (PLC)s are industrial computers designed to execute control tasks based on real-time input from sensors. They play an role in automating processes, such as opening valves, starting pumps..
– **Supervisory Control and Data Acquisition (SCADA)**: SCADA systems provide a supervisory layer that enables operators to monitor and control industrial processes remotely [10].
– **Human-Machine Interface (HMI)**: human machine interfaces (HMI) systems serve as the primary interface between operators and ICSs.
– **Remote Terminal Units (RTUs)**: remote terminal units (RTUs) serve as field devices that communicate with supervisory control and data acquisition (SCADA) systems and PLCs to collect and send process data from remote locations [29].

2.2 Cybersecurity Challenges in ICSs

With the rise of Industry 4.0 and the Industrial Internet of Things (IIoT), ICSs are now increasingly interconnected with enterprise IT networks, exposing them to cyber threats that were not considered during their initial design. This convergence of information technology (IT) and OT brings both advantages in terms of improved monitoring and control but also introduces new cybersecurity challenges [4]. Unlike traditional IT systems, where *confidentiality* of data is the top

priority, the most critical aspect of ICSs is *availability*, followed closely by *integrity* [9]. Interruptions to control systems can lead to physical damage, safety incidents, or production halts, which makes continuous and reliable operation paramount. There are several key challenges in securing ICSs:

- **Legacy Systems**: Many ICSs are built on legacy hardware and software that were not designed with cybersecurity in mind. These systems often lack the ability to receive security updates, leaving them vulnerable to exploits.
- **Proprietary Protocols**: ICSs frequently rely on proprietary communication protocols, such as Modbus, DNP3, and common industrial protocol (CIP). These protocols may lack encryption or strong authentication mechanisms, making them targets for cyber attacks.
- **Lack of Patching and Updates**: ICSs are often not updated as frequently due to concerns about disrupting critical processes. This leads to prolonged exposure to potential cyber threats.
- **Increased Connectivity**: The convergence of IT and OT has increased connectivity between industrial control networks and external systems, which introduces a wider attack surface for cyber adversaries. As more devices are connected to the Internet or enterprise networks, they become susceptible to malware, ransomware, and other external attacks.
- **Insider Threats**: In addition to external attacks, ICSs are also vulnerable to insider threats, where employees or contractors with authorised access may misuse their privileges to compromise systems. This makes robust monitoring and intrusion detection crucial [28].

2.3 Multi-layer Security: the Purdue Model

To address these challenges, ICSs security follows the Purdue Enterprise Reference Architecture (PERA), commonly referred to as the Purdue Model [4]. The model divides ICSs into different layers, each representing different operational zones within an industrial system:

- **Level 0-1: Process Control**: These levels include the physical devices and sensors that interact directly with industrial processes, such as actuators, sensors, and PLCs. Security at this level focuses on protecting the real-time control of physical processes.
- **Level 2: Supervisory Control**: This level includes HMI and SCADA systems, which provide operational oversight and process visualisation. Security concerns here include preventing unauthorised control and ensuring accurate reporting of process data.
- **Level 3: Operations Management**: At this level, the focus shifts to production management systems and databases that track operational performance. Security at this level is concerned with protecting sensitive operational data and preventing disruptions to production schedules.
- **Level 3.5: DMZ**: This level forms the boundary between the enterprise IT network and the OT network, often containing firewalls, IDSs, and other security mechanisms that help segregate the two environments.

By segmenting networks and applying security controls at each layer, the Purdue Model enhances the ability of ICSs to withstand and contain cyber attacks. Each layer is protected individually, ensuring that if one layer is compromised, the attack cannot easily spread to other parts of the system.

2.4 Intrusion Detection aided by Machine Learning

IDSss have long been a central part of cybersecurity countermeasures, particularly in information technology environments. Machine Learning (ML) has become a central focus in enhancing the detection capabilities of IDSss, both in traditional IT systems and, increasingly, in ICSs environments. Techniques such as *Decision Trees* [25], *Random Forests* [6], *Support Vector Machines* [8], and *Neural Networks* [11] have been widely applied to anomaly detection in network traffic. In IT environments, these models have shown considerable promise in detecting intrusions, malware, and other cyber threats, especially in scenarios where traditional signature-based systems fail. In ICSs, these techniques must be adapted to handle the operational constraints and real-time requirements specific to industrial systems, as well as the distinct characteristics of the process data. The complexity of modern ICSs environments, characterised by heterogeneous networks, proprietary protocols, and a mix of legacy and modern equipment, requires customised IDSss that can integrate both the network and the processing of data to detect cyberphysical threats [10].

Some research efforts have been made on ML-aided intrusion detection for ICS environments. Research by Goh et al. [12] developed a deep learning-based IDSs specifically for ICSs, leveraging process data to detect anomalies. Their work highlights the potential for deep learning to improve detection accuracy by modelling the complex relationships between sensors, actuators, and control systems. Similarly, Kravchik and Shabtai [19] employed convolutional neural networks (CNNs) to detect cyber attacks in ICSs, demonstrating how ML techniques can process high-dimensional time-series data to uncover subtle anomalies. There is also an increasing focus on hybrid approaches that combine both network and operational data. Studies have shown that combining network-based and process-based anomaly detection can significantly improve overall security posture by identifying attacks that target both network infrastructure and physical processes managed by ICSs components [16]. Konijn [17] presented a comprehensive study that integrates network traffic and process data to enhance IDSs capabilities in industrial environments. This multi-domain approach aims to detect complex, multi-stage cyber attacks that affect both network communications and physical processes. Konijn highlighted that much of the literature in ICSs security focuses either on network or physical sensor data, with few studies investigating their integration. This gap is particularly critical for detecting sophisticated attacks, such as advanced persistent threats (APTs), which often span multiple layers of the Purdue model. Shalyga et al. [26] proposed an anomaly detection system that integrates network traffic and operational data using neural networks to detect cyber attacks targeting ICSs. This multilayered approach is effective in detecting attacks that traditional network intrusion

detection systems (NIDS) might miss, such as those that directly manipulate process variables. Similarly, Jadidi et al. [16] introduced a layered IDSs designed to detect flooding attacks in ICSs environments. Their system combines network traffic analysis with physical sensor data, demonstrating the importance of multi-domain analysis for comprehensive intrusion detection. The inclusion of both data sources allows for the detection of attacks that would otherwise be missed if only one data type were considered.

While significant advancements have been made in IDSs for ICSs, the integration of ML techniques and multi-domain data remains a critical area for further research. This work seeks to contribute to the field by combining network and process data to enhance anomaly detection capabilities, building on the foundation established by earlier studies.

3   SWaT Testbed

The testbed SWaT, developed by iTrust Labs at Singapore University of Technology and Design (SUTD), is a high-fidelity ICS testbed that simulates the operation of a modern water treatment facility [20]. Commissioned in 2015 with support from the Ministry of Defence (MINDEF) of Singapore, the testbed is designed to provide researchers with a realistic environment to study cyber-physical security threats in ICSs environments. The SWaT testbed is widely used in academic and industrial research, particularly in the areas of anomaly detection and other cybersecurity solutions.

The testbed spans approximately 90 square metres and consists of six distinct stages that collectively simulate a real-world water treatment process, from raw water intake to reverse osmosis and membrane cleaning. Each stage is controlled by Programmable Logic Controllers (PLCs) and other industrial grade devices commonly found in OT environments [20]. This infrastructure closely mirrors that of modern water treatment plants, making it an ideal platform for studying the impact of cyber attacks on both physical processes and network infrastructure.

3.1   Architecture of the SWaT Testbed

The SWaT testbed is designed around six sequential stages of water treatment:

- **P1: Raw Water Supply** - pumping raw water into the system.
- **P2: Chemical Dosing** - adds chemicals for purification and water safety.
- **P3: Ultrafiltration** - filtering out particles from the water.
- **P4: De-chlorination via Ultraviolet (UV) Light** - using UV light to remove chlorine and other contaminants.
- **P5: Reverse Osmosis** - purifying water by filtering it through membranes.
- **P6: Backwash Cleaning** - cleaning the ultrafiltration membranes to prevent fouling.
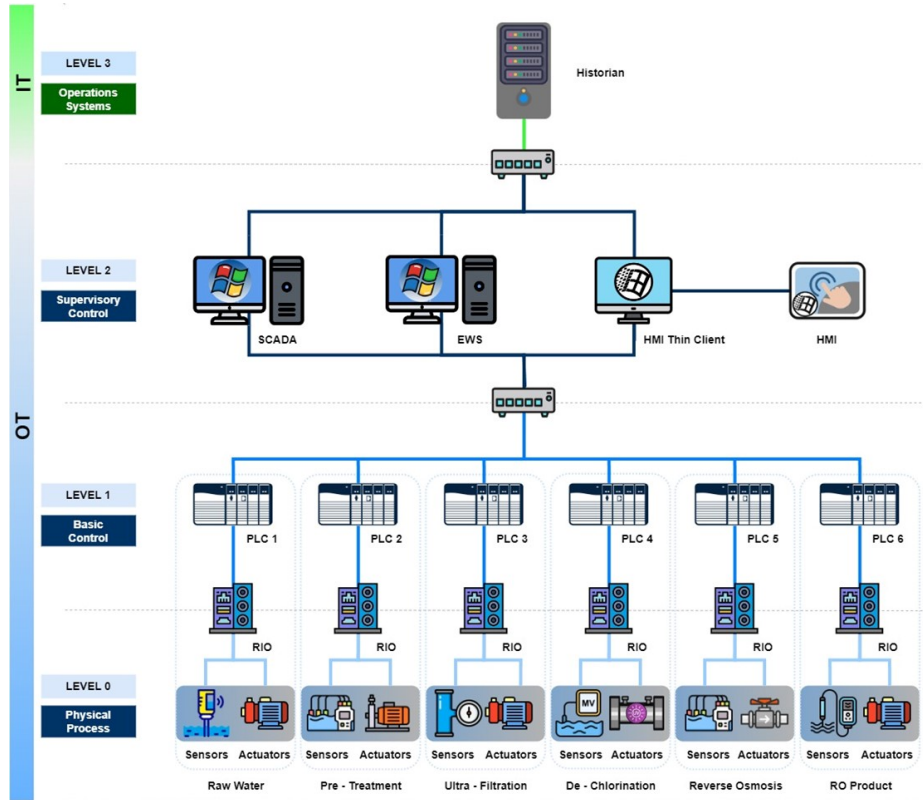
Fig. 1. Network Architecture of the SWaT Physical Process [20]

Each stage is equipped with its own sensors, actuators, and PLC to control the operation of the physical processes. The control system in the SWaT testbed mimics those in real industrial environments, where physical processes are closely monitored and controlled by a network of PLC, HMI, and SCADA systems [20]. In addition, the SWaT testbed is segmented according to the Purdue Model, with different network zones assigned to different functional areas, such as the plant floor, the control room, and the enterprise network.

3.2 Cybersecurity Research on the SWaT Testbed

The SWaT testbed has been used in numerous cybersecurity studies to evaluate the effectiveness of IDSs, intrusion prevention systems (IPS), and other defensive mechanisms in ICSs. By providing real-time access to both network traffic and physical process data, the testbed allows researchers to simulate and study the impact of various cyber attacks on critical infrastructure.

The SWaT dataset, publicly available through iTrust Labs, contains both normal operation data and data collected during a series of staged cyber attacks
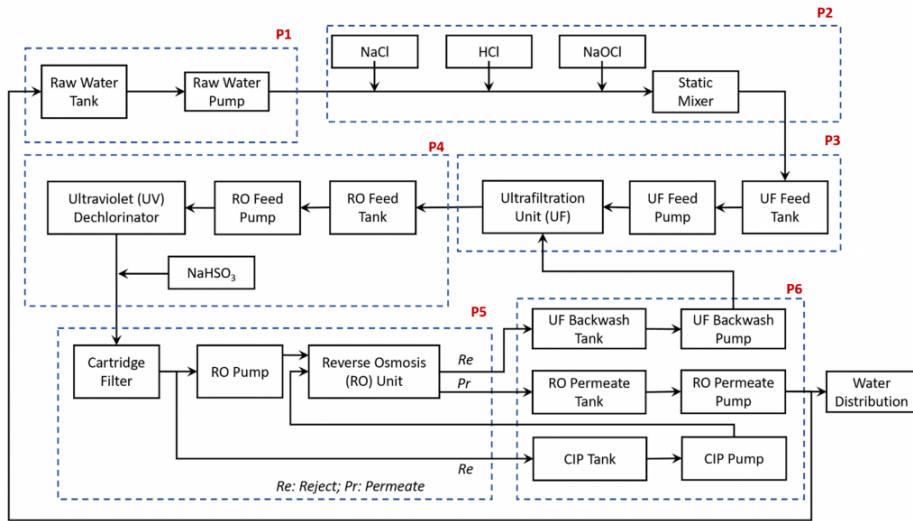
Fig. 2. Block Diagram of the SWaT Physical Process [20]

on the water treatment process. These attacks include manipulation of PLCs, sensor spoofing, and other tactics that affect both the physical and network layers of the system. The dataset includes logs of sensor readings, actuator states, and network traffic, making it an ideal resource for the development of machine learning-based IDSss [20].

Researchers have used the SWaT dataset to study a wide range of attack scenarios, including insider threats, network-based attacks, and attacks on physical processes. One of the key strengths of the SWaT testbed is its ability to provide a comprehensive view of both the network and physical layers, enabling a deeper understanding of how cyber attacks propagate through ICSs environments and affect physical processes.

In this study, the SWaT dataset is used to evaluate the performance of various machine learning models for anomaly detection in ICSs. Using both network traffic data and physical process data, this research aims to investigate the potential benefits in detecting attacks at various levels in ICSs that may go undetected by more traditional IT based IDSs. The data set provides a unique opportunity to explore the interplay between network and process anomalies, offering insight into how these systems can be protected, as well as providing valuable insight from cyber physical threats.

4   Methodology

This section outlines the dataset, preprocessing steps, machine learning models, feature extraction techniques, and the evaluation metrics used in the study.

4.1 A6 - dataset

The dataset used for this research, the A6 dataset from iTrust, is derived from the SWaT testbed, described in Section 3. Most importantly, this dataset consists of two main data streams:

1. **Network traffic data**: Captures communication between PLC and other devices, including network logs, packet details, and communication protocols. In terms of the Purdue Model 2.3, this would be data laying on layer 1 through 3 (and 3.5).
2. **Process data**: Includes sensor readings and actuator states from the various stages of the water treatment process. These data points are critical for monitoring the physical behaviour of the system. From the Purdue Model 2.3, this would be items at layer 0.

The dataset comprises normal operation data, as well as data collected during a series of staged cyber attacks, which target both the network and physical layers of the system [20]. The attacks include various forms of network intrusions, sensor spoofing, and manipulation of actuators, simulating real-world cyber-physical threats. This dual-domain dataset is ideal for testing intrusion detection systems that can monitor both network and process anomalies.

4.2 Network Data Extraction with Zeek

Zeek is an open-source tool designed for network traffic analysis, capable of dissecting network packets and producing structured logs that provide detailed insights into the behaviour of communication protocols and network traffic patterns. Zeek's flexibility in supporting custom log generation allowed us to tailor the analysis to focus on the key communication logs relevant to the SWaT testbed experiment.

In this work, we focused on three specific types of logs generated by Zeek: `cip.log`, `conn.log`, and process variable communication. These logs were selected based on their relevance to different layers of the Purdue Model, ensuring a comprehensive analysis of the interactions.

– **conn.log**: The `conn.log` provides a general overview of the network connections in the system, recording details such as source and destination IP addresses, protocol types, connection durations, and byte counts. This log gives insight into the overall network traffic, corresponding to Purdue Level 3. It is particularly useful for identifying network-level anomalies, such as unauthorised connections or unusual traffic patterns that might indicate the presence of a network-based attack.
– **cip.log**: This log captures communication using the CIP, a widely used protocol for communication between PLC and other industrial devices in the OT environment. By analysing `cip.log`, we gain visibility into the communication between critical industrial components (Purdue Level 1 and 2) and can detect deviations from normal protocol behaviour.

– **Process variable communication**: Process data, collected from sensors and actuators, corresponds to Purdue Level 0 (sensors and actuators). In this research, process data was collected alongside network data, allowing us to monitor the behaviour of physical processes. This dual-layer approach enables the detection of anomalies that may not be visible through network traffic alone, such as sensor spoofing or manipulation of actuators.

The combination of these three data sources allows us to map different layers of the Purdue Model to the corresponding logs, creating a comprehensive view of both network-level and process-level interactions. The choice of focussing on `conn.log`, `cip.log`, and process variable communication was a deliberate decision aligned with the hierarchical structure of the Purdue model 2.3 as well as the availability of data in our dataset. By having logs that reflects both the high-level network communications to low-level physical process variables we aim to capture new aspects and correlation from the ICSs. This layered approach mirrors the multilayered security strategy advocated by industrial security frameworks, such as ISA/IEC 62443 [1], and highlights the importance of monitoring all layers of ICSs to detect anomalies in the physical and network processes.

4.3   Feature Extraction from Zeek Logs

Once the Zeek logs were generated, the next step is to extract meaningful features for machine learning models. Feature extraction from the `conn.log` and `cip.log` files focused on identifying key attributes related to the communication flow between components and identifying anomalous behaviours indicative of cyber attacks. Key features extracted from these logs include:

– `conn.log`: Source and destination IPs, port numbers, duration of connections, protocols used, packet sizes, and connection status. These features help in identifying unauthorised connections, port scanning activities, or unusual traffic patterns that may suggest a Distributed Denial of Service (DDoS) attack or a network intrusion.
– `cip.log`: Specific attributes related to CIP communication, such as command types, response codes, and message sizes. These features are critical for detecting anomalies in the control communication between PLCs and sensors or actuators, which could indicate a manipulation of industrial processes.
– **Process Variable Communication**: Sensor readings (e.g., flow rates, pressure, valve states) were treated as time series data. Key features included average values, standard deviations, and abrupt changes or spikes that could indicate a manipulation of the physical system. This layer provides insight into whether the observed network anomalies correspond to actual changes in the physical process.

These extracted features formed the basis of our machine learning models, providing a combination of network-level and process-level data that enables the detection of both traditional IT-based attacks (e.g., unauthorised access, DDoS) and OT-based attacks (e.g., sensor spoofing, actuator manipulation).

4.4 Data Preprocessing

Before training the ML models, several preprocessing steps were applied to ensure the quality and usability of the dataset:

– **Handling Missing Data**: Missing or corrupted data points in both network traffic and process logs were handled through *interpolation* where possible.
– **Data Labelling**: The dataset contains semi-labelled data, with timestamps indicating known attack windows. Data within these windows were labelled as "malicious", while data outside were considered "benign". Given the lack of precise attack indicators in the dataset, this labelling was manually aligned with the attack events documented in the SWaT testbed timeline [20].
– **Normalisation**: To ensure that all features were on a comparable scale, especially for models sensitive to feature scaling (such as Support Vector Machines), min-max normalisation was applied. This step ensures that each feature's values fall within a specific range, preventing any feature from dominating the learning process due to its scale.

4.5 Feature Engineering for Anomaly Detection

One of the critical steps in this research was the extraction of meaningful features from the raw datasets. For both network traffic and process data, feature engineering was used to highlight patterns that might be indicative of cyber attacks. While it's not feasible to discuss every feature in detail, we focused on those that provided a general context for both network and process behaviour. In particular, features that did not lend themselves well to aggregation, such as specific IP address columns, were excluded. For example, calculating the most frequent IP address for a one-second interval would dilute the critical indicators. An engineer's workstation might interact with hundreds of devices within a one-second interval, but the presence of a single malicious command-and-control IP would be a much stronger indicator of an attack. Removing features such as IP source and destination allowed the models to detect general trends in traffic and process behaviours rather than specifics. Key features included:

– **Network traffic features**: Packet sizes, protocol types, connection durations, and byte counts. These features capture patterns in communication that may be altered during a cyber attack.
– **Process data features**: Sensor readings (e.g., pressure, flow rate) and actuator states (e.g., valve positions). Features such as abnormal sensor spikes or actuator state changes were identified as potential indicators of a physical process under attack.

4.6 Machine Learning Models

Several machine learning models were selected for this study based on their proven performance in intrusion detection and anomaly detection in both IT and OT domains:

– **Random Forests** [6]: A robust ensemble method that uses multiple decision trees to make predictions. Random Forests were chosen for their ability to handle mixed data types and their resistance to overfitting. They also provide feature importance rankings, which help in understanding which features contribute most to detecting anomalies.
– **Support Vector Machines (SVM)** [8]: SVMs were selected for their effectiveness in high-dimensional spaces, making them suitable for identifying complex, non-linear patterns in the dataset.
– **K-Nearest Neighbour (kNN)** [18]: A simple, yet effective classifier that predicts the class of a sample based on the majority vote of its nearest neighbours. kNN was used due to its simplicity and effectiveness in identifying outliers or anomalous instances.
– **Neural Networks (NN)** [23]: We experimented with both shallow and deep feed-forward neural networks, as well as a 1D-Convolutional Neural Network (CNN) architecture. Neural Networks were chosen for their ability to capture non-linear relationships in the data, making them suitable for detecting subtle anomalies in both network and process data.
– **Ensemble Model with Fuzzy Clustering** [27]: To leverage the diverse dataset, we implemented an ensemble model incorporating Fuzzy Clustering. This method allowed the neural network to integrate both explicit information from the original features and implicit patterns discovered through clustering.

4.7   Model Training and Testing

Each machine learning model was trained and tested using an 80/20 split of the dataset. Cross-validation was employed to ensure the models generalised well to unseen data. The models were trained on three types of datasets:

1. **Network traffic data only**.
2. **Process data only**.
3. **Combined network and process data**.

This approach allowed us to compare the performance of the models on different data types and assess the benefit of integrating both data streams for anomaly detection. For hyperparameter tuning, grid search was used to optimise parameters such as the number of trees in Random Forests, the penalty parameter C in SVM, and the number of neighbours in kNN. For neural networks, different architectures were explored, with various hidden layers and activation functions, and dropout layers were included to prevent overfitting.

4.8   Data Selection and Aggregation

To manage the large volume of data and synchronise high-frequency network traffic with process variables, we applied a time-aggregation technique as an essential step in data preparation. Entries were grouped into one-second intervals

to align with the sampling rate of the process data, enabling a unified time axis for both network and OT datasets.

Aggregating data requires balancing data integrity with manageable granularity. We used two standard aggregation techniques to simplify the data while retaining essential information: for numerical features, we calculated the mean within each interval to capture representative values; for categorical data, including those transformed via one-hot encoding, we used the mode to maintain the most frequent state within each interval. By reducing millions of raw network packets over a 3-hour 45-minute capture window into one-second summaries, this approach yielded a dataset that remains aligned with the temporal structure needed for anomaly detection while preserving critical details across network and OT events.

The aggregation also helps with noise reduction and mitigates the effects of missing values and outliers. Grouping data by intervals ensures a reliable dataset, providing consistent feature representations for machine learning - across the entire timeline. By structuring network and OT data along a common timeline, this aggregation allows us to test our hypothesis for detecting cross-layer event correlations within the ICS environment. While aggregation into one-second intervals reduces the dataset's size and complexity, it inevitably sacrifices some fine-grained detail. This trade-off between granularity and data manageability is a fine line, but in our case, necessary to enable processing and training.

4.9 Data Processing and Model Evaluation

After aggregation, several preprocessing techniques were applied to enhance data quality and optimise the model's detection capabilities. A primary challenge was the class imbalance inherent in cybersecurity datasets like SWaT, where benign instances vastly outnumber attack instances. We addressed this with synthetic minority over-sampling technique (SMOTE) [7] to balance the datasets by generating synthetic samples for the minority (attack) class, increasing the likelihood of capturing anomalous behaviour.

principal component analysis (PCA) [3] was also used to reduce the data's dimensionality. This helps simplifying model training while retaining the "significant" or *principal* features. PCA is particularly beneficial for high-dimensional datasets, as it mitigates the risk of overfitting while capturing the most impactful data points across both network and process features.

The core focus of this research is to improve the detection capabilities of intrusions across levels in ICSs. Our evaluation centres on *Recall*,

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}},$$

This focus prioritises the model's ability to detect the majority of attack instances, even at the risk of false positives. In ICSs settings, maximising recall is essential, as missing an attack could lead to severe operational or safety consequences [22]. Additionally, we calculated weighted precision by averaging pre-

cision across classes (attack and non attack) weighted by the number of true instances per class, ensuring a balanced evaluation in this unbalanced dataset.

5 Experiment and Discussion

5.1 Objective and Metric Choice

The motivation behind this study is to explore whether incorporating both network and process data can enhance intrusion detection capabilities in ICSs. Given the complex nature of ICS environments, this proof-of-concept research evaluates the potential advantages of using a combined data approach. Specifically, we focus on two key metrics: *weighted precision* and *recall*.

Weighted precision addresses the significant imbalance between attack and non-attack instances in ICS datasets, providing a more balanced view of model performance. This metric calculates precision separately for each class (attack and non attack), weighting the precision score by the number of true instances in each class. By doing so, we obtain an indicator of performance that fairly reflects the prevalence of the under-represented attack class. This measure is important for maintaining accuracy in detecting malicious activity without excessively misrepresenting benign instances.

Recall, or sensitivity, is equally important, particularly in ICS environments where undetected attacks can have severe operational or safety consequences. This metric emphasizes the model's ability to identify attack instances (true positives) while minimising the likelihood of false negatives. Given our research goal of improving detection for ICS security, recall provides an ideal measure of a model's capacity to catch attack behaviours accurately across a complex, multi-layered environment. Together, these metrics help to assess whether combining data sources meaningfully contributes to more reliable detection capabilities.

Weighted precision and recall were specifically chosen for this proof-of-concept study due to the unique challenges faced in ICSs environments, where detecting even rare anomalies is crucial. This focus on these metrics helps illustrate the models' potential in minimising false negatives within the constraints of this preliminary setup.

5.2 Experiment setup

Each model was trained and tested on three datasets to evaluate the comparative and complementary detection capabilities of network-only, process-only, and combined datasets:

1. Network traffic data only: This dataset consists solely of network communication logs (e.g., `conn.log` and `cip.log`), capturing information such as packet sizes, connection times, and IP addresses. Network data offers insight into traffic anomalies but is often limited in detecting attacks affecting physical processes.

14

2. Process data only: This dataset includes sensor readings and actuator states that directly represent physical process behaviour. Process data is valuable for detecting disruptions in the physical layers of the system, such as sensor spoofing or actuator manipulation, yet may miss network-centric attack patterns.
3. Combined network and process data: Merging network and process data provides a broader perspective on system state, integrating both traffic and operational insights. By testing the combined dataset, this study seeks to understand if additional features from both data streams improve the model's ability to detect coordinated or multistage attacks that span both network and process levels.

This approach allows us to observe the distinct contributions of each dataset and to evaluate the benefit of integrating data sources. The results obtained from the singular datasets can be viewed as our *baseline results* and could reflect the capabilities from an IDS in our testbed who only observes the network data, whereas in the *combined network and process data* section, we take a look at at the data combinations we have available and their "new" capabilities.

5.3 Baseline Results

The baseline results, summarised in Figure 3, show the distribution of attack recall for each dataset without applying SMOTE. To not include the SMOTE versions in the baseline results is due their overfitting nature, which was observed in preliminary tests where SMOTE introduced non-representative synthetic samples, particularly in the *conn.log* and *cip.log* datasets. Among the datasets, *cip.log* showed the weakest performance, likely due to noise present in this dataset.
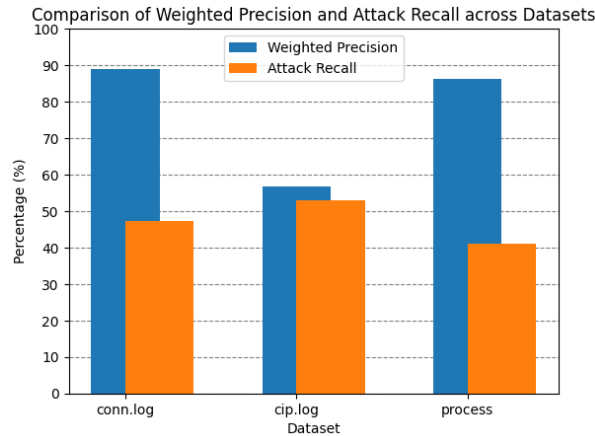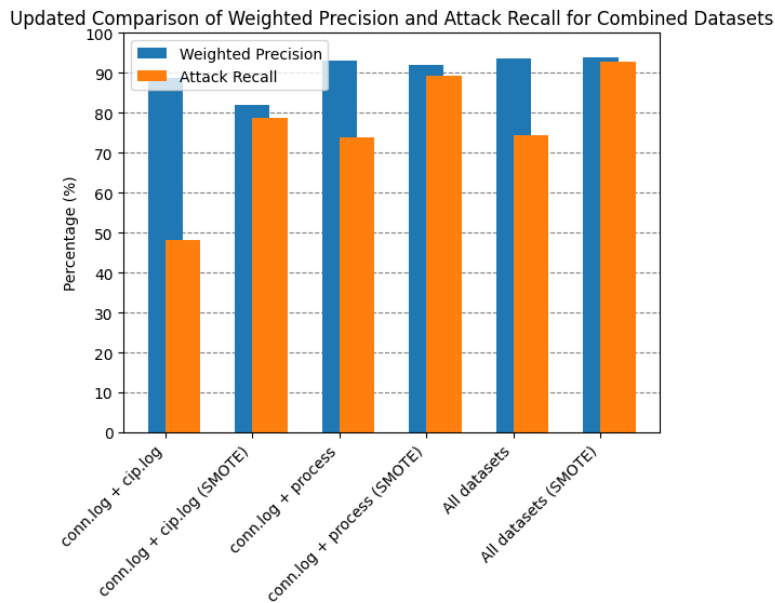

Fig. 3. Histogram showing the distribution of attack recall for baseline models

5.4 Combined Dataset Results

When combining datasets (*conn.log* + *cip.log*, *conn.log* + *process data*, and *conn.log* + *cip.log* + *process data*), the model performance improved, as illustrated in Figure 3. The expanded range of features in the combined datasets appears to enable the models to capture the system's state more effectively, leading to better detection of anomalies and attack instances across the dataset.

Applying SMOTE to the combined datasets further enhanced performance by addressing class imbalances. Unlike in the baseline, where SMOTE introduced overfitting, the richer feature set in the combined datasets seems to allow SMOTE to generate more representative samples along a *broader* feature-range, resulting in more robust learning and improved recall and precision.

Fig. 4. Histogram showing the distribution of attack recall for combined models

The combined datasets, particularly the combination of network traffic and process data, significantly improved model detection performance, as evidenced by increases in recall and precision metrics.

6 Conclusion

This research highlights that combining network traffic data with operational process data presents a promising opportunity to enhance and broaden the scope

of cyber attack detection in ICSs environments. The results suggest that additional data available in OT systems can play a valuable role in improving the depth and scope of anomaly detection. However, these findings are limited to the specific context of the SWaT dataset and the proof-of-concept nature of this study. Future work should extend these investigations across varied ICSs datasets and incorporate more advanced machine learning techniques to better assess the generalisability and robustness of integrated data approaches in detecting complex cyber threats within industrial settings.

References

1. Industrial Communication Networks – part 3-3: System security requirements and security levels, https://webstore.iec.ch/publication/6028.
2. Sklearn Document, https://scikit-learn.org, [Online; accessed Apr. 2024]
3. Abdi, H., Williams, L.J.: Principal component analysis. Wiley interdisciplinary reviews: computational statistics **2**(4), 433–459 (2010)
4. Ackerman, P.: Industrial Cybersecurity: Efficiently monitor the cybersecurity posture of your ICS environment. Packt Publishing Ltd (2021)
5. Alexander, O., Belisle, M., Steele, J.: Mitre att&ck for industrial control systems: Design and philosophy. The MITRE Corporation: Bedford, MA, USA **29** (2020)
6. Breiman, L.: Random forests. Machine Learning **45**(1), 5–32 (2001).
7. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: Smote: synthetic minority over-sampling technique. Journal of artificial intelligence research **16**, 321–357 (2002)
8. Cortes, C., Vapnik, V.: Support vector networks. Machine Learning **20**(3), 273–297 (1995).
9. DNV: Cyber priority 2023. Tech. rep., DNV AS, accessed: 1. Desember 2023
10. Galloway, B., Hancke, G.P.: Introduction to industrial control networks. IEEE Communications surveys & tutorials **15**(2), 860–880 (2012)
11. Gamage, S., Samarabandu, J.: Deep learning methods in network intrusion detection: A survey and an objective comparison. Journal of Network and Computer Applications **169**, 102767 (2020).
12. Goh, J., Adepu, S., Tan, M., Lee, Z.S.: Anomaly detection in cyber physical systems using recurrent neural networks. In: 2017 IEEE 18th International Symposium on High Assurance Systems Engineering (HASE). pp. 140–145 (2017)
13. Garcia-Teodoro, P., Diaz-Verdejo, J., Maciá-Fernández, G., Vázquez, E.: Anomaly-based network intrusion detection: Techniques, systems and challenges. computers & security **28**(1-2), 18–28 (2009)
14. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer Series in Statistics, Springer New York (2009).
15. Hutchins, E.M., Cloppert, M.J., Amin, R.M., et al.: Intelligence-driven computer network defense informed by analysis of adversary campaigns and intrusion kill chains. Leading Issues in Information Warfare & Security Research **1**(1), 80 (2011)
16. Jadidi, Z., Foo, E., Hussain, M., Fidge, C.: Automated detection-in-depth in industrial control systems. The International Journal of Advanced Manufacturing Technology **118**(7), 2467–2479 (2022)
17. Konijn, J.P.: Multi-domain Cyber-attack Detection in Industrial Control Systems. Master's thesis, University of Twente (2022)

18. Kumar, S., Choudhary, G., Kumar, Y.: An analysis of the k-nearest neighbor technique for the intrusion detection system. In: 2017 IEEE International Conference on Computing and Communication Technologies for Smart Nation (IC3TSN). pp. 325–329 (2017).

19. Kravchik, M., Shabtai, A.: Detecting cyberattacks in industrial control systems using convolutional neural networks, Proceedings of the 2018 Workshop on Cyber-Physical Systems Security and Privacy, pp. 72-83 (2018)

20. iTrust Labs: Swat technical report, https://itrust.sutd.edu.sg/itrust-labs_datasets/dataset_info/

21. Liu, X., Zhu, J., Li, L.: 1d convolutional neural networks and applications: A comprehensive review. Journal of Neural Networks (2019).

22. Mishra, P., Varadharajan, V., Tupakula, U., Pilli, E.: A detailed investigation and analysis of using machine learning techniques for intrusion detection. IEEE Communications Surveys & Tutorials **21**(1), 686–728 (2019).

23. Mishra, P., Varadharajan, V., Tupakula, U., Pilli, E.S.: A detailed investigation and analysis of using machine learning techniques for intrusion detection. IEEE Communications Surveys and Tutorials **21**(1), 686–728 (2019).

24. Patcha, A., Park, J.M.: An overview of anomaly detection techniques: Existing solutions and latest technological trends. Computer networks **51**(12), 3448–3470 (2007)

25. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann (1993)

26. Shalyga, D., Filonov, P., Lavrentyev, A.: Anomaly detection for water treatment system based on neural network with automatic architecture optimization (2018)

27. Smith, J.W., Wilson, M.L.: An intelligent hybrid system for pattern recognition: Integrating fuzzy c-means clustering with neural networks. Neural Networks Journal (1998).

28. Stouffer, K., Pease, M., Tang, C., Zimmerman, T., Pillitteri, V., Lightman, S.: Guide to operational technology (ot) security. National Institute of Standards and Technology: Gaithersburg, MD, USA (2022).

29. Sullivan, D., Luiijf, E., Colbert, E.J.: Components of industrial control systems. Springer (2016)

30. Singapore University of Technology, S.U., (SUTD), D.: iTrust Labs. https://itrust.sutd.edu.sg/itrust-labs-home/itrust-labs_swat/, accessed: Accessed: 6. Oct. 2023