Management and Production Engineering Review
Volume 16 • Number 4 • December 2025 • pp. 1–14
DOI: 10.24425/mper.2025.157213

Enhancing SCADA Security: AI Based Approaches for Attack Detection

Belayadi DJAHIDA[1], Djeghlouf ASMAA[2], Lebkara HAITHEM[3], Ouar NARIMANE[4]

[1] *National Higher School of Cybersecurity, Basic Training Department, Algeria*
[2] *National Higher School of Advanced Technologies, Industrial Engineering and Maintenance Department, Algeria*
[3] *University of Hassiba Ben Bouali, Electronics Department Chlef, Algeria*

Received: 24 April 2025
Accepted: 10 October 2025

Abstract
In the industrial sector, Supervisory Control and Data Acquisition (SCADA) systems are essential for managing Industrial Internet of Things (IIoT) networks. However, these systems have become increasingly exposed to cyberattacks targeting the communication layers embedded in industrial processes. Such vulnerabilities can cause severe disruptions in manufacturing and production environments. The ongoing digitalization of Industrial Control Systems (ICS) has further amplified these risks, emphasizing the need for robust security mechanisms such as Intrusion Detection Systems (IDS). This research aims to develop a high-precision AI-based IDS capable of protecting SCADA systems from evolving cyber threats. To achieve this, three categories of machine learning algorithms were evaluated: Deep Learning models (CNN, RNN, LSTM), Boosting algorithms (XGBoost, GBoost, AdaBoost), and classical methods (RF, DT, KNN). Extensive experiments were conducted using two benchmark SCADA datasets, WUSTL-IIoT-2018 and WUSTL-IIoT-2021. The results demonstrated outstanding detection performance, with all models achieving accuracy rates above 99.91%. Specifically, RF, DT, KNN, and XGBoost reached perfect accuracy (100%) on the WUSTL-IIoT-2018 dataset, while XGBoost, LSTM, and CNN achieved 99.99% accuracy on WUSTL-IIoT-2021. Additional evaluation metrics, including precision, recall, and F1-score, confirmed the robustness of the models. The findings highlight the potential of AI-driven IDS solutions to enhance the security and resilience of industrial SCADA infrastructures.

Keywords
SCADA, Industrial IoT, Attack Detection System, Cybersecurity, Machine Learning, Deep Learning, Boosting Algorithms, Industrial Control Systems.

Introduction

In recent years, initiatives like Industry 4.0 and the Internet of Things (IoT) have emerged, offering promising prospects to change the manufacturing landscape. Industry 4.0 introduces a revolutionary wave of smart manufacturing, integrating AI-driven elements like digital marketplaces and the seamless convergence of the physical and cyber worlds. Whereas, IoT refers to the interconnection of physical devices through networks, enabling the collection and exchange of data about their operation and environment.

Together, these concepts are transforming industries by enabling real-time monitoring, improved decision-making, and enhanced operational efficiency. IoT technologies are now utilized across various sectors including industry, healthcare, agriculture, and energy. A key subset, the Industrial Internet of Things (IIoT), focuses specifically on industrial applications and plays a central role in the digital transformation of this sector (Lu et al., 2019). IIoT enables the networked connectivity of instruments, sensors, and devices in manufacturing environments, allowing for data collection, analysis, and control. Many traditional industrial companies are now embracing digital transformation through IIoT. To ensure sustainability and security, Industrial Control Systems (ICS) must be capable of operating and monitoring critical infrastructure (Stouffer et al., 2015).

One such system is the Supervisory Control and Data Acquisition (SCADA) system, which is used to optimize energy usage, reduce waste, and monitor operations across various industrial automation environments (Patel et al., 2019). SCADA systems typically follow a layered architecture, composed of key elements such as the Master Terminal Unit (MTU), Remote Terminal Unit (RTU), Human-Machine Interface (HMI), and Programmable Logic

***Corresponding author**: Belayadi Djahida – National Higher School of Cybersecurity Sidi AbdAllah, Algiers, phone: (+213) 556-24-16-09 e-mail: djahida.belayadi@enscs.edu.dz*

Controllers (PLCs). The MTU serves as the central supervisory computer and is often equipped with an HMI, where real-time data collected from RTUs and PLCs is visualized and acted upon (Stouffer et al., 2015). RTUs are responsible for collecting and storing real-time sensor data, which is transmitted to the MTU when needed. PLCs, connected to edge devices, gather sensor data and control actuators. SCADA systems use either wired or wireless communication channels for data transmission. These channels are crucial for integrating and monitoring complex industrial processes, enabling information retrieval and process management across multiple remote sites.

SCADA systems rely on various communication technologies to enable data transmission between field devices and control centers. Commonly used communication methods include Ethernet, which offers high-speed, wired connectivity, cellular networks, suitable for remote or mobile installations, satellite communications, used in geographically isolated locations; and industrial wireless protocols such as WirelessHART and ISA100.11a, which are tailored for harsh industrial environments and provide flexibility and scalability (Gungor & Hancke, 2009; Riggins & Wamba, 2015). The choice of communication technology depends on factors such as latency requirements, network topology, environmental conditions, and security constraints.

While SCADA systems continue to evolve from isolated architectures to networked systems, they also become more vulnerable to cyber threats. This transition, though beneficial, introduces new risks, as the increased connectivity opens up more attack surfaces (Chen et al., 2017). Cyberattacks can lead to significant consequences, including disruptions to essential public services such as electricity, gas, and water, as well as damage to critical infrastructure posing serious risks to public safety.

As SCADA systems become more interconnected, traditional cybersecurity measures often prove insufficient. New and more sophisticated cyber threats continue to emerge. To combat this, Artificial Intelligence (AI) can be leveraged to improve threat detection, reduce response times, and enhance the overall protection of critical infrastructure (Xu et al., 2020).

The increasing efficiency of communication over the internet in the previous decades paired with technological development has made it easier to carry out cyberattacks like Denial of Service (DoS), Phishing, Hijacking, SQL Injection, Man in The Middle (MiTM), Reconnaissance, and Password attacks targeting SCADA systems (Karami & Shamsi, 2019).

This research focuses on identifying cyber threats within SCADA systems to improve the security of these vital infrastructures. Nine different AI algorithms were applied: Adaptive Boosting (AdaBoost), Extreme Gradient Boosting (XGBoost), Gradient Boosting (GBoost), Long Short-Term Memory (LSTM), Convolutional Neural Networks (CNN), and Recurrent Neural Networks (RNN), Random Forest (RF), Decision Tree (DT), k-Nearest Neighbor (KNN) to detect cyberattacks on SCADA systems. These algorithms were tested on two distinct SCADA datasets, and their performance in attack detection was carefully evaluated. The hyperparameters of each algorithm were optimized to assess its effectiveness in identifying and mitigating potential threats.

The structure of this paper is as follows: Section 2 discusses SCADA systems and cybersecurity. Section 3 reviews the related works in the field. Section 4 provides an overview of the datasets, AI methods employed, and the performance metrics used for evaluation. Section 5 presents the experimental results and compares them with existing literature. The final section concludes the study with a summary of key findings.

SCADA Systems and Cybersecurity

This section provided an overview of SCADA systems, detailing their components, the cybersecurity measures in place, and the potential threats they may face.

SCADA Systems

A typical SCADA system architecture (Fig. 1) consists of several vital components that collaborate cohesively to enable real-time observation and management of industrial operations (Patel et al., 2019). At the center is the Master Terminal Unit (MTU), the primary supervisory computer that orchestrates the system's operations. It communicates with Remote Terminal Units (RTUs) and Programmable Logic Controllers (PLCs), which collect data from field instruments and control actuators.

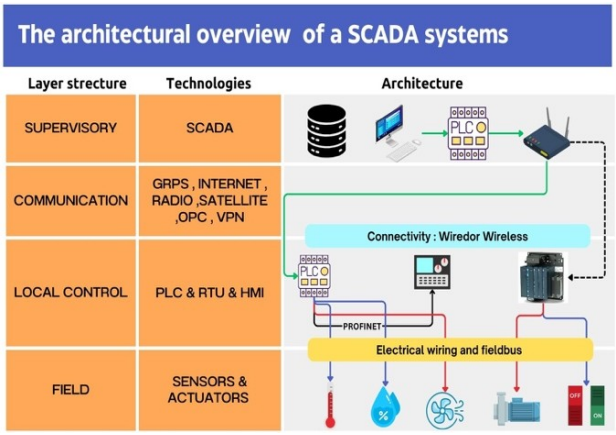

Fig. 1. The architectural overview of a SCADA system

RTUs gather real-time data from sensors, switches, and remote devices deployed at field locations, transmitting this data to the MTU. PLCs are responsible for controlling actuators and field devices used in physical processes. The Human-Machine Interface (HMI) provides operators with a visual interface to monitor system status and perform control actions (Daneels & Salter, 1999).

While SCADA systems offer vital advantages in terms of operational efficiency, their increasing complexity and interconnectivity with other systems expose them to a variety of cybersecurity threats. These vulnerabilities make it critical for industries to adopt enhanced security measures such as AI-based Intrusion Detection.

Systems (IDS) to safeguard these essential infrastructures from malicious attacks.

Cybersecurity and Threats in SCADA Systems

SCADA systems are critical to monitoring and controlling a variety of industrial processes, but their networked nature makes them vulnerable to cyberspace risks. Common attack types include Denial of Service (DoS) attacks, Phishing, SQL Injection attacks and Man-in-the-Middle (MiTM) attacks (Tab. 1) and can damage the validity of critical industrial processes (Xu et al., 2020).

Table 1
Some types of attacks involved in SCADA systems

Type of Attack	Description
Denial of Service (DoS)	Disrupts the availability of SCADA systems by overwhelming them with traffic.
Phishing	Social engineering attacks to gain access to sensitive SCADA system information.
Man-in-the-Middle (MiTM)	Inserting malicious SQL queries to manipulate or access SCADA database systems.
Reconnaissance	Collecting data about SCADA system vulnerabilities before launching an attack.
Password Attacks	Attempting to guess or crack passwords to gain unauthorized access to SCADA systems.
Privilege Escalation	Gaining elevated access levels in SCADA systems to execute unauthorized commands.

Because SCADA systems are often connected to external networks, they are inherently exposed to cyber risks. A single successful attack can result in the loss of control, system failure, or data breaches. To mitigate these risks, advanced cybersecurity solutions are required. In particular, intelligent systems like AI-powered IDS can significantly enhance the security posture by detecting anomalies, identifying attacks, and ensuring continuous operation of essential infrastructure (Karami & Shamsi, 2019).

Literature review

SCADA systems security has emerged as a critical issue in recent years because of the rising number of cyberattacks against essential infrastructures. Reconnaissance attacks act as initial steps that can lead to more destructive actions including DoS, data manipulation and system hijacking. The need to detect these attacks has led to increased research efforts. These studies have frequently used ML and DL-based methods for attack detection. Some of these works are summarized below.

A study referenced as (Smith & Brown, 2018) reveals how SCADA systems become vulnerable because they consist of combined hardware and software elements from multiple manufacturers which complicates their security. SCADA systems exhibit vulnerability to cyberattacks because their complexity increases risk when they connect to external networks. Security experts have identified phishing and hijacking attacks along with reconnaissance as frequent threats to SCADA systems. This research identifies cyberattacks effects on essential infrastructure systems such as energy grids alongside transportation networks and water supply systems.

Other works, like those by Tesfahun et al. (2016), investigate cybersecurity vulnerabilities in SCADA systems. In contrast, researchers such as Krishnan et al. (2019) focused on testbeds for vulnerability assessments and penetration testing. Moreover, studies like those of (Kalech, 2019) employ temporal pattern recognition to detect cyber-attacks, while more recent work, like (Dehlaghi-Ghadim et al., 2023), introduces anomaly detection datasets for industrial control systems.

Study of Zainudin et al. (2022) focuses on enhancing the classification of DDoS attacks, particularly targeting low-latency requirements in IIoT systems. The authors propose a combination of XGBoost for feature selection and a hybrid CNN-LSTM approach for classification. This hybrid method aims to improve the

accuracy rate and reduce complexity, making it suitable for real-time, low-latency IIoT environments. The work utilizes the CICDDoS2019 dataset for training and evaluation.

In Qaiser et al. (2023) work, researchers use four different datasets to classify DDoS attacks in IIoT systems. They apply six different ML algorithms and report average F1-scores between 88% and 94%, showing improved detection performance across varied attack scenarios.

The paper reported by Trivedi et al. (2023) explores the effects of adversarial ML attacks on ICS. They use several machine learning algorithms (SVM, DT, and CNN) to model and analyze system behavior under attack conditions. The results showed that system performances are significantly impacted by these attacks, with models demonstrating varying degrees of vulnerability under adversarial conditions. The study reports that classification accuracy with adversarial attacks dropped to 87% on average. Williams & Zhang (2020) conducted a study to detect reconnaissance attacks represented by port scanning and network probing activities. The study proposes a method to analyze the traffic patterns and behaviors of intruders using ML techniques to identify suspicious activities. They report a precision of 91% and an F1-score of 89%, demonstrating the effectiveness of behavior-based traffic classification.

Recent research has seen a significant shift towards utilizing DL techniques to bolster the security of SCADA systems. Notably, works by Gao, X. et al. (2023) and Wu et al. (2023) emphasize the remarkable progress in applying deep learning to this domain, while Yang & Chen (2019) suggest that deep learning approaches offer great promise for addressing security challenges in SCADA systems, positioning them as a key avenue for future security enhancements.

Morris et al. (2023) at Mississippi State University's SCADA Security Lab focused on the most widely used communication protocols. They aimed to detect cyberattacks and reduce their impact by developing security mechanisms based on neural network methods to enhance the protection of SCADA systems against potential threats. Their model achieved an F1-score above 90%, outperforming traditional methods in intrusion detection tasks.

Research conducted by Wang et al. (2022) developed a stacked deep learning model for detecting cyberattacks, which effectively identifies suspicious behaviors and distinguishes them from normal actions. The approach achieved a precision of 93.2% and a recall of 94.1%, outperforming traditional shallow ML models such as NB, RF and SVM in terms of intrusion detection performance.

Work performed by Jmila & Houda (2022) examined shallow classifiers commonly used in IDS, noting their continued popularity due to their simplicity and reliability. They evaluated how these classifiers, including AdaBoost, DT and RF, fared against adversarial attacks, alongside a deep learning network for comparison. The results revealed that the effectiveness of each classifier varied based on the type of attack, with AdaBoost achieving the highest F1-score (88%) for DoS detection while CNN slightly outperformed others for reconnaissance detection.

In their study Gao et al. (2023) examined the detection of cyberattacks on SCADA systems, specifically addressing attacks that are either temporally correlated or uncorrelated. Advanced DL techniques are used, including feedforward neural networks (FNN) and Long Short- Term Memory (LSTM) models, to identify these threats. Their combined FNN-LSTM approach achieved an F1-score of 91.7% and was effective in detecting both attack types.

Other research proposed a deep learning-based approach called Omni SCADA (Gao et al., 2020), it combines the benefits of feedforward neural network (FNN) and long-short term memory (LSTM) for detecting temporally uncorrelated and correlated attacks. Their method demonstrated strong performance in identifying and preventing attacks, with average precision and recall values above 92%.

Recently, much research has been done on developing intrusion detection mechanisms for SCADA systems. For instance, see some relevant survey papers (Radoglou-Grammatikis & Sarigiannidis, 2019; Zeng & Zhou, 2018; Rakas et al., 2020; Quincozes et al., 2021; Cui et al., 2020; Rahman et al., 2025).

In this work, we utilized the "WUSTL-IIoT-2018" and "WUSTL-IIoT-2021" datasets to evaluate the security of SCADA systems. These datasets simulate reconnaissance attacks, enabling the development of high-performance detection models. To identify and classify cyber threats, we implemented and tested nine AI algorithms, including AdaBoost, XGBoost, GBoost, LSTM, CNN, RNN, RF, DT, and KNN.

Materials & Methods

Datasets Description

This study utilizes two publicly available SCADA datasets: WUSTL-IIoT-2018 and WUSTL-IIoT-2021, to evaluate the performance of various AI-based models for cyberattack detection in industrial systems. Both datasets were collected in simulated SCADA environments and contain labeled traffic that includes both normal and malicious activity, making them suitable benchmarks for intrusion detection system (IDS) evaluation.

WUSTL-IIOT-2018

The WUSTL-IIOT-2018 dataset (in St. Louis, 2018) consists of 7,037,983 records, including 6,634,581 normal traffic instances and 403,402 abnormal traffic instances. The abnormal traffic includes five types of reconnaissance-related attacks: address scan, port scan, exploit attempts, device identification, and aggressive mode device identification. Designed as a generic SCADA system representation, the dataset is commonly used for evaluating IDS performance in identifying system vulnerabilities. It contains seven key continuous features extracted from both normal and attack traffic. These features (Tab. 2) include source port, total packets, total bytes, source packets, destination packets, source bytes, and traffic label (indicating attack or not). A correlation matrix (Fig. 2) was also generated to analyze the relationships between these features and to identify potential redundancies.

Table 2
Selected Features of the WUSTL-IIOT Datasets

Feature	Signification	Type	Description
Sport	Source Port	Integer	Port number of the source
TotPkts	Total Packets	Integer	Total transaction packet count
TotBytes	Total Bytes	Integer	Total transaction bytes
SrcPkts	Source Packets	Integer	Source/Destination packet count
DstPkts	Destination Packets	Integer	Destination/Source packet count
SrcBytes	Source Bytes	Integer	Source/Destination transaction bytes
Target	Traffic Status	String	The status of traffic (with or without attack)

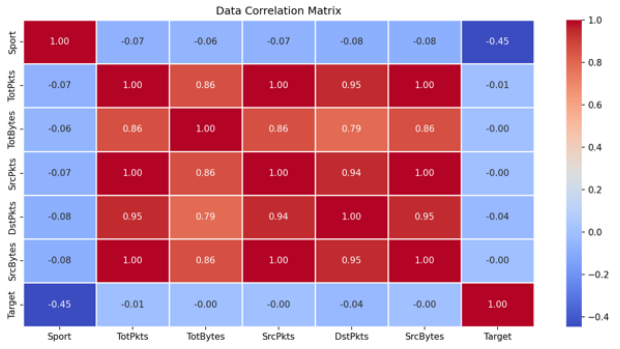

Fig. 2. The correlation between the features in the WUSTL-IIOT-2018 dataset

WUSTL-IIOT-2021

The WUSTL-IIOT-2021 (in St. Louis, 2021) dataset comprises 1,194,464 instances composed of 1,107,448 normal traffic records as well as 87,016 abnormal traffic records. The WUSTL-IIOT-2021 database consists of four attack types: command injection attacks combined with DoS attacks as well as reconnaissance attacks and backdoor/other attacks. The dataset offers complete information about numerous SCADA system attacks thus allowing researchers to test security solutions for contemporary industrial control systems.

Both datasets exhibit significant class imbalance, with the number of normal traffic instances far outweighing the number of attack instances. For example, in WUSTL-IIoT-2018, only about 5.7% of the total records are labeled as attacks, while in WUSTL-IIoT-2021, abnormal records constitute approximately 7.3% of the data. This imbalance can negatively impact model performance, especially for traditional classification metrics like accuracy, which may appear high even if the model fails to detect most attacks. To address this, class imbalance was taken into consideration during model training and evaluation. We applied stratified sampling during the train-test split to preserve the class distribution. In addition, performance was assessed using metrics more suitable for imbalanced datasets such as precision, recall, F1-score, false positive rate (FPR), and area under the precision-recall curve (AUPR), to ensure a reliable evaluation of the models' ability to detect minority-class (attack) instances.

Artificial Intelligence Methods

Three categories of AI techniques were selected for detecting SCADA reconnaissance attacks: deep learning methods, boosting algorithms and classical machine learning models.

Deep Learning Methods

Convolutional Neural Networks: CNN (LeCun et al., 2015) is a type of feedforward neural network that employs convolutional filters to maintain the connectivity between neurons. It draws inspiration from the visual cortex of animals and finds applications in various domains such as image and signal recognition, recommendation systems, and natural language processing (NLP). CNNs typically require structured input data in the form of multi-dimensional arrays. For image data, the input format is usually a 3D tensor representing the height, width, and color channels (e.g., RGB) of an image. For time-series or signal data, the

input is typically a 2D array, where each row represents a time step or a feature. CNNs are composed of convolution layers, which are generally linear and are followed by the application of an activation function (e.g., ReLU), which performs convolution calculations to extract important features. Subsequently, pooling layers are used to reduce the dimensionality of the data. Finally, the classification layers (fully connected) are used to predict the output.

Recurrent Neural Networks: RNNs (Rumelhart et al., 1986) are neural networks designed to handle sequential data by incorporating memory through feedback loops. Unlike feedforward networks, RNNs can retain information from previous inputs, making them suitable for time-series data, speech recognition, and language translation. However, RNNs often suffer from the vanishing gradient problem, limiting their ability to learn long-term dependencies.

Long Short-Term Memory networks: LSTMs (Hochreiter & Schmidhuber, 1997) are derivatives of RNN. They can learn and memorize addictions over a long period. LSTMs thus retain the information stored over the long term. They are especially useful for predicting time series, they remember previous entries. In addition to this use case, LSTMs are also used to compose musical notes and recognize voice.

Boosting Methods

Gradient Boosting: Gboost (Friedman, 2001) functions as an ensemble methodology which develops models sequentially by adding new models to reduce errors from preceding models. A decision tree model fits into the residuals of existing models to enhance predictive accuracy through the process. The loss minimizing capability of Gradient Boosting through gradient descent makes this approach effective for regression and classification problems.

Extreme gradient Boosting: XGboost (Chen & Guestrin, 2016) was introduced in 2014. It is another type of boosting algorithm, constructed as an ensemble of gradient boosting decision trees using a sequential learning approach. Adding a regularization parameter that lessens each regression tree's susceptibility to dataset outliers, which improves the gradient boosting algorithm. With this approach, a weight value is assigned to each data value in the database, defining the likelihood that the value will be chosen for additional examination by a decision tree. Afterwards, the sample class is predicted by adding the trees sequentially. Every tree seeks to recoup the discrepancy between the target and the previous ensemble of trees forecast.

Adaptive Boosting: Ada boost (Freund & Schapire, 1996) Freund Yoav and Schapire Robert were the first searchers who introduced and built this model in 1995. It is one of the boosting algorithms that was named accordingly because it adapts adaptively to the flaws of the weak hypotheses that WeakLearn returns, in contrast to other algorithms. It may combine any number of base-learners and re-uses the same training set. This means that many classifiers of AdaBoost are trained one after the other. The efficiency of previously taught classifiers is the basis for training each new one.

Classical Machine Learning Methods

Random Forest: RF (Cutler et al., 2012) is an ensemble learning method that grows a multitude of decision trees during training and combines their outputs, typically by majority voting for classification or averaging for regression, with the goal of improving predictive accuracy and alleviating overfitting. The method implements bootstrap aggregation (bagging) in combination with random feature selection at each split, thus increasing the model's robustness and diversity.

Decision Tree: DT (Basak, 2006) is a non-parametric supervised learning model that recursively partitions the input space based on feature thresholds, thus forming a hierarchical structure in the shape of a tree, where internal nodes represent decision-making criteria and leaf nodes represent predicted outcomes. The model has high interpretability and allows a clear explanation of the decision-making process but can be prone to overfitting without proper pruning or regularization methods.

K-Nearest Neighbor: KNN (Rovatti et al., 1995) is an instance-based learning method used for both classification and regression tasks. It classifies a new observation by determining the majority class among its k nearest neighbors in the feature space, where closeness is measured using different distance metrics such as Euclidean or Manhattan distance. The performance of the algorithm relies heavily on the choice of the parameter k and feature normalization, thus necessitating careful parameter tuning and preprocessing for optimal results.

Performance Evaluation Metrics

To assess model performance, a variety of evaluation metrics were used, including accuracy (Acc), precision (P), recall (R), and F1-score (F1) (Tab. 3). These metrics provide insight into the models' ability to correctly identify both normal and malicious traffic, especially in the presence of class imbalance. Additionally, Receiver Operating Characteristic (ROC) curves and Precision-Recall (PR) curves were plotted to evaluate classification performance under different

threshold conditions. Metric formulas are based on values derived from the confusion matrix.

Table 3
The Performance Metrics

Metric	Formula
Acc	$(TP+TN)/(TP+TN+FP+FN)\times 100\%$
FPR	$FP/(TN+FP)\times 100\%$
P	$TP/(TP+FP)$
R	$TP/(TP+FN)$
F1	$2\times(P\times R)/(P+R)$

Methodology

This study aims to detect various cyber-attacks on SCADA systems using nine AI- based techniques: CNN, RNN, LSTM, XGBoost, GBoost, AdaBoost, RF, DT and kNN.

The methodology begins with data preprocessing, which involves cleaning the datasets, selecting relevant features, and normalizing the data based on feature range analysis. Following preprocessing, the data is divided into training (70%) and testing subsets (30%), ensuring that the models are trained on one portion of the data and evaluated on another to measure their generalization capability. For model evaluation, we employed three-fold cross-validation (3-fold CV). This process splits the training data into three subsets, where each subset is used once as a validation set while the remaining two subsets are used for training. This cross-validation technique ensures that each model is tested on different portions of the data, providing a more reliable estimate of its performance. The models' performance is evaluated using multiple metrics, including accuracy, precision, recall, and F1-score. An overview of the complete workflow is illustrated in Fig. 3.

Results and Discussion

Each SCADA dataset was normalized using the MinMaxScaler technique to ensure consistent scaling across the data. After normalization, the dataset was split into two subsets: 70% for training, while the remaining 30% was reserved for testing. This partitioning strategy was employed to effectively train the models and evaluate their performance. A detailed overview of this data distribution is presented in the Tab. 4.

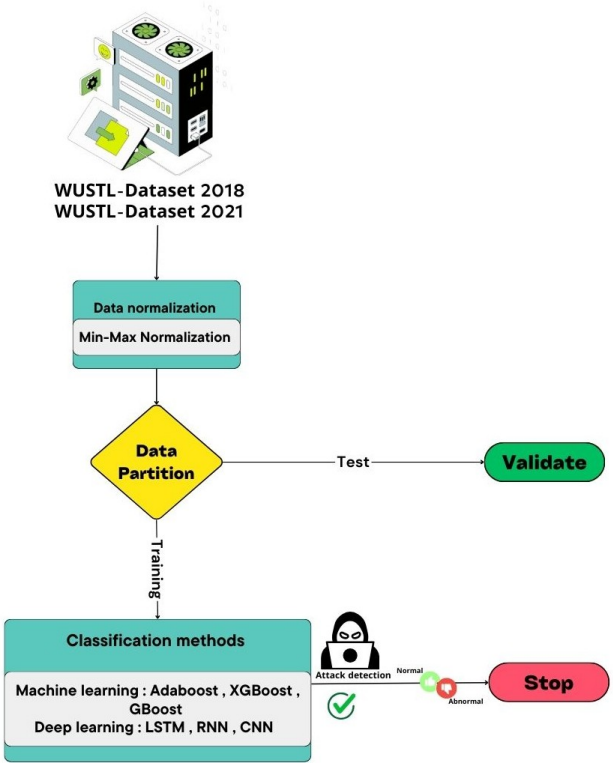

Fig. 3. The flowchart of our methodology

Table 4
Statistics of the Datasets

Dataset	Class	Train / Test / Total Samples
WUSTL-IIOT 2018	0 (normal)	4,644.485 / 1,990.096 / 6,634.581
	1 (abnormal)	282.103 / 121.299 / 403.402
	Total	4,926,588 / 2,111,395 / 7,037.983
WUSTL-IIOT 2021	0 (normal)	775.152 / 332.296 / 1,107.448
	1 (abnormal)	60.972 / 26.044 / 87.016
	Total	836.124 / 358.340 / 1,194.464

Hyperparameters

A set of hyperparameters was carefully selected for optimal performance in the models. In the current study, this involved a thorough tuning process whereby each parameter was carefully adjusted to balance training effectiveness, convergence speed, and model generalization. The hyperparameters are defined in Tab. 5.

Table 5
Summary of Hyperparameter Settings

Category	Parameter	Value
Deep Learning Models		
DL Models	Optimizer Loss Function Batch Size Epochs Hidden Activation Output Activation	Adam Binary Cross-Entropy 64 10 ReLU Sigmoid
Boosting Models		
AdaBoost	Estimators Random State Weak Learner	50 42 DecisionTreeClassifier (max_depth=1)
XGBoost	Estimators Random State Objective/Evaluation	50 42 Binary Logistic / Log-Loss
Gradient Boosting	Estimators Learning Rate Loss Function Max Depth	50 0.1 Log-Loss 3
Traditional ML Models		
Random Forest	n_estimators Criterion Max Depth Min Samples Split	200 Gini None 10
Decision Tree	Criterion Max Depth Min Samples Split	Entropy None 10
KNN	Metric n_neighbors Weights	Manhattan 6 Distance

Model Optimization

We used RandomizedSearchCV as an optimization process, which allows us to efficiently explore a variety of hyperparameter combinations for each model. By randomly sampling from the hyperparameter space, while setting a specific number of iterations and employing three-fold cross-validation, this method helps to lower computational costs while still finding configurations that are close to optimal. This strategy not only enhances model accuracy and generalization by reducing the risk of overfitting but also guarantees reproducibility by maintaining a fixed random state. In general, this effective hyperparameter optimization plays a crucial role in ensuring strong performance of our models.

Results

The variations between the models are illustrated by the confusion matrices in Fig. 4. For example, in the confusion matrix that we had for DL methods in Wustl-IIot-2018, LSTM recorded the lowest false positives (330) and false negatives (45), showing its consistency in minimizing misclassification. CNN also performed well with only 353 false positives and 106 false negatives, confirming its ability to learn temporal dependencies in IIoT traffic. However, RNN had the highest misclassification rate, with 599 false positives and 520 false negatives, indicating potential difficulties in distinguishing normal and attack traffic compared to the other models. These results overall suggest that CNN and LSTM are more suitable for IIoT intrusion detection, while RNN may have to be fine-tuned to be made more robust.

The results shown in Tab. 6 highlight the performance of boosting, deep learning, and traditional machine learning models on the WUSTL-IIOT-2018 and WUSTL-IIOT-2021 datasets. In the 2018 dataset, boosting models performed exceptionally well, with XGBoost achieving a perfect test accuracy of 100%, followed closely by GBoost at 99.99% and AdaBoost at 99.98%. These models also exhibited low false positive rates, such as XGBoost's rate of 0.0252. Among the deep learning models, CNN had a slight edge over RNN and LSTM, reaching a test accuracy of 99.97%, which underscores its effectiveness in capturing spatial patterns. Traditional machine learning models such as Random Forest (RF) and Decision Tree (DT) also delivered strong results, with test precision near perfection (99.99%–100%).

The results for the WUSTL-IIOT-2021 dataset closely mirror those of the WUSTL- IIOT-2018 dataset. In both cases, boosting models and deep learning techniques present strong results in terms of Accuracy, Precision, Recall, and F1-score. Traditional machine learning models also showed impressive performance in both datasets. This consistency highlights the strong generalization abilities of the models used and affirms their reliability in identifying anomalies in SCADA systems across various datasets. These findings indicate

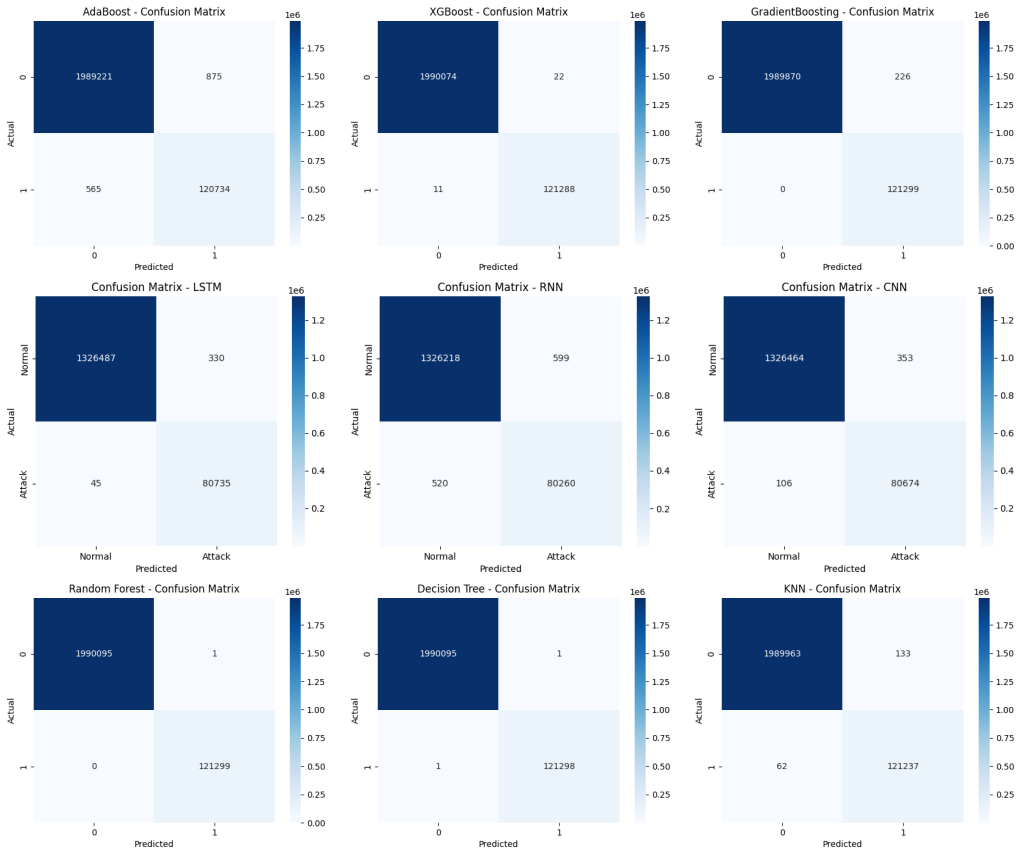

Fig. 4. Confusion matrices for the 9 algorithms of the study for WUSTL-SCADA-2018

that boost models, especially XGBoost, offer the best combination of accuracy, precision, recall and F1 score with minimal false positive rate (FPR) for detecting anomalies in IIot Systems.

In addition, to strengthen the presented results, a comparative analysis of ML and DL models enrolled in the WUSTL-IIOT-2018 and WUSTL-IIOT-2021 datasets is illustrated in Fig. 5 and Fig. 6. The accuracy, precision, recall, and F1-score metrics of each model are plotted in the shown histograms, that visually demonstrate the performance of SCADA cyber threat detection models.

In addition, we conducted another analysis of the three models of each category using additional performance metrics. In the following, we present a discussion based on the ROC curve and the Precision-Recall (PRC) curve on the Wustl-Scada-2018 dataset (Fig. 7 and Fig. 8) for DL Learning models.

In our experiment with the WUSTL-SCADA-2018 dataset, the ROC curves show that both the CNN and LSTM models achieve an AUC of 1.00, representing perfect classification with no false positives and false negatives, while the RNN model achieves a slightly lower AUC of 0.99. The ROC curves of LSTM and

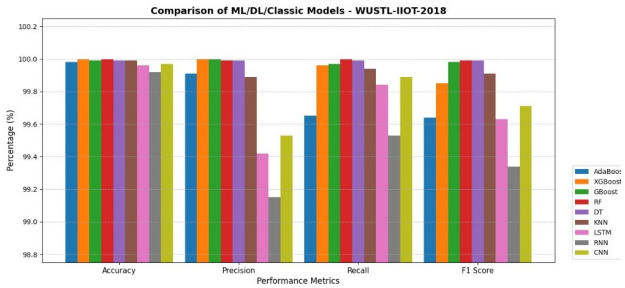

Fig. 5. Model Performance comparison histogram for WUSTL-SCADA-2018

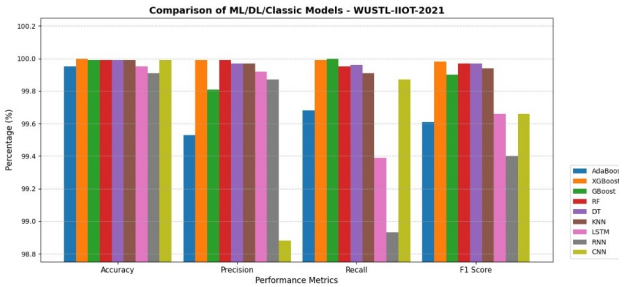

Fig. 6. Model Performance comparison histogram for WUSTL-SCADA-2021

Table 6
Performance Results

Dataset	Method	Training Acc (%)	Test Acc (%)	Precision	R	F	FPR
	AdaBoost	99.98	99.98	0.9991	0.9965	0.9964	0.0864
	XGBoost	100	100	1.0000	0.9996	0.9985	0.0252
	GBoost	99.99	99.99	1.0000	0.9997	0.9998	0.0678
	RF	100	100	0.9999	1.0000	0.9999	0.066
WUSTL-IIOT-2018	DT	100	99.99	0.9999	0.9999	0.9999	0.3333
	KNN	100	99.99	0.9989	0.9994	0.9991	0.1430
	LSTM	99.95	99.96	0.9942	0.9984	0.9963	0.5833
	RNN	99.91	99.92	0.9915	0.9953	0.9934	0.4612
	CNN	99.95	99.97	0.9953	0.9989	0.9971	0.5370
	AdaBoost	99.95	99.95	0.9953	0.9968	0.9961	0.0342
	XGBoost	100	99.99	0.9999	0.9999	0.9998	0.0279
	GBoost	99.99	99.99	0.9981	1.0000	0.9990	0.0240
	RF	99.99	99.99	0.9999	0.9995	0.9997	0.0192
WUSTL-IIOT-2021	DT	100	99.99	0.9997	0.9996	0.9997	0.3333
	KNN	99.99	99.99	0.9997	0.9991	0.9994	0.1429
	LSTM	99.99	99.95	0.9992	0.9939	0.9966	0.2303
	RNN	99.91	99.91	0.9987	0.9893	0.9940	0.3491
	CNN	99.99	99.99	0.9888	0.9987	0.9966	0.3416

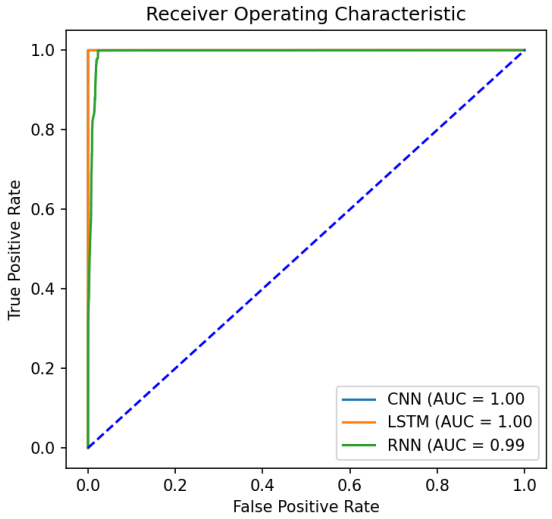

Fig. 7. ROC chart

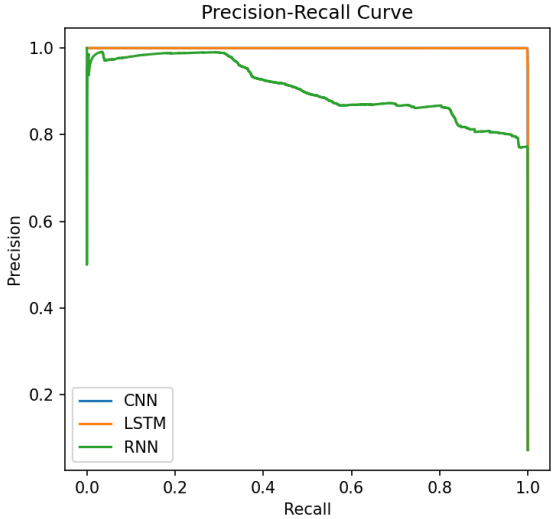

Fig. 8. PRC chart

CNN are close to perfect, converging towards the top-left point of the graph, whereas the curve of RNN, while being very good, has very low misclassifications.

Similarly, the Precision-Recall curves show that the LSTM model has precision 1.0 for all values of recall, and the CNN model has a highly balanced performance with high recall and stable precision. Conversely, the RNN model shows a consistent

drop in accuracy with a corresponding rise in recall, showing a slight bias towards false positives. Taken together, these findings attest that all three models are extremely competent for anomaly detection in SCADA systems, with CNN and LSTM models having a slight advantage over the RNN model.

Discussion

To benchmark our results, we compared our model performances with recent studies in the field of SCADA anomaly detection (Tab. 7 & Tab. 8).

Tab. 7 and Tab. 8 present a comparison of our study's performance metrics with those found in recent literature on SCADA anomaly detection. Our XGBoost, CNN, LSTM, and RF models consistently outperformed or matched the highest scores in accuracy, precision, recall, and F1-score reported in the literature. Compared to Al-Abassi et al. (2020), where CNN achieved 98.21% accuracy and 70.50% F1-score, our CNN achieved 99.95% accuracy and 99.71% F1-score. Our models also outperformed GRU, SVM, and MLP models used in prior works (Yalçın et al., 2024; Huma et al., 2021; Alzahrani & Aldhyani, 2023).

These comparisons emphasize the effectiveness and robustness of our proposed methodology and model selection for securing SCADA environments against a wide range of cyber threats.

Table 7
Comparison of Model Performance with recent studies for WUSTL-IIOT-2018

Reference	Model	Acc (%)	P (%)	R (%)	F1 (%)
	Modified DT	99.99	99.99	00.88	99.93
	RF	99.99	99.93	99.88	99.93
Yalçın et al., 2024	AdaBoost	99.98	99.97	99.80	99.88
	XGB	99.99	99.99	99.82	99.91
	GB	99.99	99.99	99.91	99.95
	RF	99.57	99.67	99.57	99.59
Huma et al., 2021	CNN	92.74	89.25	99.89	98.71
	LSTM	95.76	95.07	95.76	95.64
	RF	95.80	95.40	99.70	95.60
Al-Abassi et al., 2020	kNN	94	94.60	99.50	94.20
	MLP	79.20	88.20	97.20	83.20
	GRU	99.75	99.76	99.43	99.50
Alzahrani & Aldhyani, 2023	CNN-GRU	98.18	99.10	98.95	98.85
	RF	96.81	–	–	–
Zolanvari et al., 2019	DT	94.26	–	–	–
	KNN	93.44	–	–	–
	SVM	80.84	–	–	–
	LR	66.20	–	–	–
	FNN	98.95	77.22	64.06	68.48
Dina et al., 2023	CNN	98.21	88.54	66.51	70.50
	AdaBoost	99.95	99.53	99.68	99.61
	XGBoost	100	99.99	99.99	99.98
	GBoost	99.99	99.81	100	99.90
	LSTM	99.99	99.92	99.39	99.66
This study	RNN	99.91	99.87	98.93	99.40
	CNN	99.99	98.88	99.87	99.66
	RF	99.99	99.99	99.95	99.97
	DT	100	99.97	99.96	99.97
	KNN	99.99	99.97	99.91	99.94

Table 8
Comparison of Model Performance with recent studies for WUSTL-IIOT-2021

Reference	Model	Acc (%)	P (%)	R (%)	F1 (%)
Chen et al., 2017	ANN	98.40	99.57	98.02	98.97
	Modified DT	89.00	87.31	86.47	86.47
	RF	87.31	87.31	87.31	87.31
Yalçın et al., 2024	AdaBoost	86.47	86.47	86.47	86.47
	XGB	86.47	86.47	86.47	86.47
	GB	87.31	87.31	87.31	87.31
Dina et al., 2023	GSFTNN ResNet RNN LSTM	98.54 97.70 94.22 95.98	98.70 98.10 93.64 96.30	98.42 97.54 93.73 95.78	98.61 97.89 94.38 96.17
Alzahrani & Aldhyani, 2023	GRU CNN-GRU	99.93 99.98	99.95 99.98	99.94 99.98	99.95 99.98
Tamy et al., 2019	Naive Bayes SVM J48	94.20 94.20 99.20	94.60 94.50 99.20	94.20 94.20 99.20	93.00 92.60 99.10
This study	AdaBoost XGBoost GBoost LSTM RNN CNN RF DT KNN	99.98 100 99.99 99.95 99.91 99.95 100 100 99.99	99.91 100 100 99.15 99.15 98.53 100 99.99 99.89	99.65 99.96 99.97 99.53 98.53 99.89 99.99 99.99 99.94	99.64 99.85 99.88 99.34 99.34 99.71 99.97 99.99 99.91

Conclusion

Our study demonstrates that a comprehensive AI-based approach can significantly enhance the detection of cyberattacks in SCADA systems. These systems, essential for managing critical industrial processes, are increasingly exposed to sophisticated threats due to the growing interconnectivity of industrial environments. By evaluating nine different algorithms – ranging from deep learning techniques (CNN, RNN, LSTM) to boosting methods (AdaBoost, XGBoost, GBoost) and traditional machine learning models (RF, DT, KNN) – we achieved detection metrics (accuracy, precision, recall, and F1-score) consistently exceeding 99.90% on both the WUSTL-IIoT-2018 and WUSTL-IIoT-2021 datasets. Our findings reveal that boosting methods and classical models, particularly XGBoost, RF, and DT, deliver nearly flawless performance, establishing a new benchmark compared to existing studies. While deep learning models also achieved strong results, slight variations in false positive rates indicate that further optimization is required for deployment in real-world industrial contexts. The robustness and generalization capability of these models demonstrate their potential to significantly strengthen SCADA system security. As a continuation of this work, future research will explore a novel hybrid approach that combines the strengths of RF and XGBoost to improve detection stability and interpretability, while addressing overfitting and supporting real-time adaptation to evolving cyber threats in industrial settings.

References

Al-Abassi, A., Karimipour, H., Dehghantanha, A., & Parizi, R. M. (2020). An Ensemble Deep Learning-based CyberAttack Detection in Industrial Control System. *Proceedings of the IEEE.*

Alzahrani, A., & Aldhyani, T. H. H. (2023). Design of Efficient Based Artificial Intelligence Approaches for Sustainable of Cyber Security in Smart Industrial Control System. *Sustainability*, *15*(10), 8076. DOI: 10.3390/su15108076

Basak, J. (2006). Online adaptive decision trees: Pattern classification and function approximation. *Neural Computation*, *18*(9), 2062–2101.

Chen, T., & Guestrin, C. (2016). XGBoost: A scalable tree boosting system. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 785–794. DOI: 10.1145/2939672.2939785

Chen, T., Zhang, H., & Li, W. (2017). Security challenges and solutions in SCADA systems: A survey. *Journal of Computer Security*, *25*(1), 1–30. DOI: 10.3233/JCS-160771

Cui, L., Qu, Y., Gao, L., Xie, G., & Yu, S. (2020). Detecting False Data Attacks Using Machine Learning Techniques in Smart Grid: A Survey. *Journal of Network and Computer Applications*, *102808*.

Cutler, A., Cutler, D.R., & Stevens, J.R. (2012). Random forests. *Ensemble Machine Learning: Methods and Applications*, 157–175.

Daneels, A., & Salter, W. (1999). What is SCADA? *International Conference on Accelerator and Large Experimental Physics Control Systems, 1999, Trieste, Italy*

Dehlaghi-Ghadim, A., Moghadam, M.H., Balador, A., & Hansson, H. (2023). Anomaly Detection Dataset for Industrial Control Systems. *ArXiv Preprint*, *2305.09678*.

Dina, A.S., Siddique, A.B., & Manivannan, D. (2023). A Deep Learning Approach for Intrusion Detection in Internet of Things Using Focal Loss Function. *Internet of Things*, *22*, 100699.

Freund, Y., & Schapire, R.E. (1996). Experiments with a new boosting algorithm. *Proceedings of the Thirteenth International Conference on Machine Learning (ICML-96)*, 148–156.

Friedman, J.H. (2001). Greedy function approximation: A gradient boosting machine. *Proceedings of the 13th International Conference on Neural Information Processing Systems (NIPS-01)*, 1026–1034.

Gao, J., & colleagues. (2023). Detection of Temporally Correlated and Uncorrelated Attacks in SCADA Systems Using FNN-LSTM Model. *Journal of Industrial Cybersecurity*, *12*(4), 102–115.

Gao, J., Gan, L., Buschendorf, F., Zhang, L., Liu, H., Li, P., Dong, X., & Lu, T. (2020). Omni SCADA Intrusion Detection Using Deep Learning Algorithms. *IEEE Internet of Things Journal*, *8*(2), 951–961.

Gao, X., & colleagues. (2023). Enhancing SCADA System Security Using Deep Learning Techniques. *Journal of Industrial Cybersecurity*.

Gungor, V.C., & Hancke, G.P. (2009). Industrial wireless sensor networks: Challenges, design principles, and technical approaches. *IEEE Transactions on Industrial Electronics*, *56*(10), 4258–4265. DOI: 10.1109/TIE.2009.2015754

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. DOI: 10.1162/neco.1997.9.8.1735

Huma, Z.E., Latif, S., Ahmad, J., Idrees, Z., Ibrar, A., Zou, Z., Alqahtani, F., & Baothman, F. (2021). A Hybrid Deep Random Neural Network for Cyberattack Detection in the Industrial Internet of Things. *IEEE Access*, *9*, 55595–55605. DOI: 10.1109/ACCESS.2021.3071766

in St. Louis, W.U. (2018). *WUSTL-IIOT-2018: Industrial Internet of Things Dataset.*

in St. Louis, W.U. (2021). *WUSTL-IIOT-2021: Industrial Internet of Things Dataset.*

Jmila, M., & Houda, A. (2022). Evaluating the Robustness of Shallow Classifiers in Intrusion Detection Systems Against Adversarial Attacks. *Journal of Cybersecurity and Machine Learning*, *15*(3), 250–265.

Kalech, M. (2019). Cyber-Attack Detection in SCADA Systems Using Temporal Pattern Recognition Techniques. *Computers & Security*, *84*, 225–238.

Karami, M., & Shamsi, S. (2019). A survey of cyberattacks on SCADA systems and their countermeasures. *Journal of Computer Networks and Communications*, *2019*, 1–17. DOI: 10.1155/2019/7472493

Krishnan, S., & Wei, M. (2019). SCADA Testbed for Vulnerability Assessments, Penetration Testing and Incident Forensics. *Proceedings of the 7th International Symposium on Digital Forensics and Security.*

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. DOI: 10.1038/nature14539

Lu, Y., Xu, X., & Wang, L. (2019). The industrial Internet of Things: A survey. *Computer Networks*, *148*, 258–277. DOI: 10.1016/j.comnet.2018.11.014

Morris, T., & colleagues. (2023). Investigating Security Vulnerabilities and Cyberattacks in SCADA Systems Using Neural Network Methods. *Mississippi State University SCADA Security Lab Research*

of Automation, I.S. (2022). *ISA100 Wireless Standard.*

Patel, V., Patel, K., & Patel, K. (2019). A review on SCADA systems and their applications in industrial automation. *International Journal of Engineering and Technology*, *7*(4), 62–68. DOI: 10.7763/IJET.2019.V7.867

Qaiser, G., Chandrasekaran, S., Chai, R., & Zheng, J. (2023). Classifying DDoS Attack in Industrial Internet of Services Using Machine Learning. *Proceedings of the 15th International Conference on Computer and Automation Engineering (ICCAE)*, 546–550.

Quincozes, S.E., Albuquerque, C., Passos, D., & Mosse, D. (2021). A Survey on Intrusion Detection and Prevention Systems in Digital Substations. *Computers and Networks*, *184*, 107679.

Radoglou-Grammatikis, P.I., & Sarigiannidis, P.G. (2019). Securing the Smart Grid: A Comprehensive Compilation of Intrusion Detection and Prevention Systems. *IEEE Access*, *7*, 46595–46620.

Rahman, M.M., Al Shakil, S., & Mustakim, M.R. (2025). A survey on intrusion detection system in IoT networks. *Cyber Security and Applications*, *3*, 100082.

Rakas, S.V.B., Stojanovic, M.D., & Markovic-Petrovic, J.D. (2020). A Review of Research Work on Network-Based SCADA Intrusion Detection Systems. *IEEE Access*, *8*, 93083–93108.

Riggins, F.J., & Wamba, S.F. (2015). Research directions on the adoption, usage, and impact of the Internet of Things through the use of Big Data analytics. *Proceedings of the 48th Hawaii International Conference on System Sciences (HICSS)*, 1531–1540. DOI: 10.1109/HICSS.2015.186

Rovatti, R., Ragazzoni, R., Kovacs, Z.M., & Guerrieri, R. (1995). Adaptive voting rules for k-nearest neighbors classifiers. *Neural Computation*, *7*(3), 594–605.

Rumelhart, D.E., Hinton, G.E., & Williams, R.J. (1986). Learning representations by backpropagating errors. *Nature*, *323*(6088), 533–536. DOI: 10.1038/323533a0

Smith, J., & Brown, T. (2018). Vulnerabilities in SCADA Systems: A Survey. *Journal of Industrial Security*, *5*(2), 123–135.

Stouffer, K., Falco, J., & Scarfone, K. (2015). *Guide to Industrial Control Systems (ICS) Security*. National Institute of Standards and Technology (NIST). https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-82r2.pdf

Tamy, S., Belhadaoui, H., Rabbah, M., Rabbah, N., & Rifi, M. (2019). Select the best machine learning algorithms for prediction and classification of intrusions using kdd99 intrusion detection dataset. *Indian Journal of Science and Technology*, *12*(37), 1–6.

Tesfahun, A., & Bhaskari, D.L. (2016). A SCADA Testbed for Investigating Cyber Security Vulnerabilities in Critical Infrastructures. *Automatic Control and Computer Sciences*, *50*, 54–62.

Trivedi, S., Tran, T.A., Faruqui, N., & Hassan, M.M. (2023). An Exploratory Analysis of Effect of Adversarial Machine Learning Attack on IoT-enabled Industrial Control Systems. *Proceedings of the International Conference on Smart Computing and Application (SCA)*, 1–8.

Wang, W., Harrou, F., Bouyeddou, B., Senouci, S.M., & Sun, Y. (2022). Cyber-Attacks Detection in Industrial Systems Using Artificial Intelligence-Driven Methods. *International Journal of Critical Infrastructure Protection*, *38*, 100542.

Williams, A., & Zhang, L. (2020). Detection of Reconnaissance Attacks in SCADA Systems Using Network Traffic Analysis. *IEEE Transactions on Industrial Informatics*, *10*(4), 1827–1836.

Wu, Y., & colleagues. (2023). Advancements in Deep Learning for SCADA System Protection. *Journal of Cybersecurity and Automation*.

Xu, Z., Li, Y., & Zhang, H. (2020). Artificial intelligence for cybersecurity in critical infrastructure protection: A review. *Future Generation Computer Systems*, *106*, 719–734. DOI: 10.1016/j.future.2019.12.024

Yalçın, N., Çakır, S., & Ünaldı, S. (2024). Attack Detection Using Artificial Intelligence Methods for SCADA Security. *IEEE Internet of Things Journal*. DOI: 10.1109/JIOT.2024.3447876

Yang, J., & Chen, L. (2019). Deep Learning Approaches for Enhancing SCADA System Security. *International Journal of Critical Infrastructure Protection*, *22*, 15–28.

Zainudin, A., Ahakonye, L.A.C., Akter, R., Kim, D.-S., & Lee, J.-M. (2022). An efficient hybrid-dnn for ddos detection and classification in software-defined iiot networks. *IEEE Internet of Things Journal*, *10*(10), 8491–8504.

Zeng, P., & Zhou, P. (2018). Intrusion Detection in SCADA System: A Survey. In *Intelligent Computing and Internet of Things* (pp. 342–351). Springer.

Zolanvari, M., Teixeira, M.A., Gupta, L., Khan, K.M., & Jain, R. (2019). Machine Learning-Based Network Vulnerability Analysis of Industrial Internet of Things. *IEEE Internet of Things Journal*, 6(4), 6822–6834. DOI: 10.1109/JIOT.2019.2918437