

Adversarial Attack Detection in Industrial Control Systems Using LSTM-Based Intrusion Detection and Black-Box Defense Strategies

Motaz Abdulaziz Almedires¹, Ahmed Elkhailil² , Mohammed Amin³ 

¹ College of Computer Sciences and Information Technology, King Faisal University, Saudi Arabia

² Computer and Information Engineering, Qilu Institute of Technology, Jinan, 250200, Shandong, China

³ Fellowship Researcher, INTI International University, Nilai 71800, Malaysia

ARTICLE INFO

Article History

Received: 15-03-2025

Revised: 01-05-2025

Accepted: 01-05-2025

Published: 05-05-2025

Academic Editor:

Prof. Youakim Badr

Vol.2025, No.3

DOI:

<https://doi.org/10.63180/jcsra.thestap.2025.3.2>



ABSTRACT

In industrial control systems (ICS), neural networks are increasingly being utilized to detect intrusions. The term ICS refers to a group of controlling technology and associated equipment that includes the devices, systems, networks, and controllers that are used to manage and/or execute manufacturing processes. Each ICS is developed to successfully handle work digitally and operates differently depending on the business. ICS devices and procedures are now found in practically every industry sector and key infrastructure, including production, transportation, power, and treatment plants. To avoid detection, attackers who aim to inflict harm on an ICS may resort to techniques such as adversarial examples to mask their attacks. ICS-based autoregressive intrusion detection systems (IDSs) are the focus of this study because of the unique issues that arise when being attacked. The attacker here is an LSTM-based IDS that can compromise a ICSs subset of sensors. In the wild cyber-physical attacks take place in ICSs that are masked from the IDS by the attacker manipulating data provided to it. Automation of ICS intrusion detection has become more flexible and efficient thanks to the growth and use of IDSs based on machine learning. Adversarial machine learning (AML), a term coined to describe cyberattacks on learning models, has been formed developed in response to the advent of the IDS. In ICSs, such attacks can have disastrous repercussions if the IDS is bypassed. Delay in attack detection could lead to damage to infrastructure, financial loss, and even human life. In this study we are proposing a defense study method that have been effective in combatting adversarial threats to ICSs and to assess adversarial attacks successfully in real-world circumstances. We are proposing a security solution IDS which can detect an adversarial attack on the industrial control system. We were able in this study to detect a black box attack by conducting DDoS attack scenario trained by black box adversarial attack in the ICS environment and use data from an ICS to train a classification model and test the ability to detect cyber intrusions in a similar context using IDS.

Keywords: Industrial Control Systems (ICSs), Adversarial Attack Detection, LSTM, Intrusion Detection, Black-Box Defense Strategies.

How to cite the article

Almedires, M. A., Elkhailil, A., & Amin, M. (2025). Adversarial Attack Detection in Industrial Control Systems Using LSTM-Based Intrusion Detection and Black-Box Defense Strategies. Journal of Cyber Security and Risk Auditing, 2025(3), 4–22. <https://doi.org/10.63180/jcsra.thestap.2025.3.2>

List of Acronyms

ICS	industrial control system
IOC	indicator of compromise
IDS	intrusion detection system
S-IDS	Sequence-aware intrusion detection system
OT	operation technology
SOC	security operation center
IT	information technology
AML	adversarial machine learning
DDoS	distributed denial-of-service
MITM	man in the middle
PCAP	packet capture
LSTM	long short-term memory

1. Introduction

The adversarial threats to deep learning systems are well-known. Machine learning systems can be tricked by attackers who make minor adjustments to the data they receive. A great deal of effort has been invested in improving offensive and defensive capabilities. The majority of research on adversarial machine learning, however, focuses on the picture domain, despite the fact that problems in other domains must be considered [1]. To defend machine learning systems that are used for security purposes, adversarial examples come into play. Machine learning techniques regularly outperform in the detection of cyber-attacks. However, susceptibility to adversarial instances makes adaptive attackers an urgent threat.

Industrial Control System (ICS) intrusion detection is a field where machine learning algorithms are simultaneously being investigated and offered by companies as security solutions; therefore, we look at the problem of hostile instances. This study focuses on ICS adversarial vulnerabilities in the context of restrictions. ICS sensors and actuators that can be hacked by an attacker and replaced with data from an intrusion detection system (IDS) are examined in this investigation [2]. An IDS can be easily compromised if the attacker has access to a significant amount of data. Circumventing detection with as few compromised sensors and actuators as possible is a difficult problem to solve.

Industrial control systems (ICSs) cover a wide range of industrial process control systems and components. ICSs are in charge of genuine data collecting, network management, and automated process control and administration [3]. Financial services, shipping, treatment plants, industrial, and energy generation and transmission are just a few of the numerous industries that have utilized ICSs extensively. They also have a direct impact on the economy because they are part of the nation's basic infrastructure. As computer and Internet technology become more interconnected, ICSs are becoming more intelligent and open [4].

ICS security has been a major public concern in recent years, and the number of cyber assaults on ICSs is expanding rapidly. Iran's Natanz nuclear enrichment facility was the target of the notorious Stuxnet malware assault in 2010, which took control of several key components and caused an abnormal acceleration of the facility's uranium-enriched centrifuge, ultimately leading to its destruction [4]. Because of this, the plant had to be shut down. By gaining access to Ukraine's power grid control center via a VPN, Black Energy disrupted the power supply by tampering with control instructions for a relay and cutting the circuit. DDoS attacks were also conducted against the system's networks and control software to prevent the system's monitoring method from detecting fault conditions and then restoring the electricity supply network [5]. at once. Dr. Jason Staggs demonstrated how to physically connect to unmanned wind turbines, i.e., to steal power in the United States at Black Hat 2017 [6].

The goal of an intrusion detection system (IDS) is to automatically detect hostile activities on a network. Gathering and analyzing data from numerous computer system components, such as internet activity, audit trails, system logs, and other important factors of the network, is part of checking for system security issues. ICS security also involves the deployment

of intrusion detection systems (IDSs) [6]. There is a lot of effort being done in this field right now, both in academia and in industry, to develop IDSs for ICSs. As a result, numerous ICS intrusion detection systems are being developed. This work discusses and suggest an alternative categorization of ICS IDS that takes into account the ICS's unique properties in order to advance ICS-specific intrusion detection research [7].

The study's motivation is to analyze adversarial attacks in the industrial control systems and the ability to train adversarial attacks in the operation technology environment and detect cyber intrusion attacks. With the buildup of massive quantities of data, the progression of computational power, and the innovative thinking and evolvement of machine learning approaches and structures, artificial intelligence (AI) techniques such as machine vision, language processing, and automated driving have been fully implemented and tried to apply all over the globe in the online world [1].

AI is on the verge of making history for humanity. Machine learning (ML) techniques, specifically, have a considerable influence on traditional computer security research [2]. Attackers may use ML to improve the accuracy of their assaults, in addition to its applications in construct different harmful detections and attack identification systems. Many disciplines in which ML is applied, from machine learning to information security, have been demonstrated to be vulnerable to adversarial assault risks in recent research.

1.1 Problem Statement

ICSs have become a popular target for attackers, based on several security issues. One of the most pressing international challenges is how to protect the security of ICSs. Many security issues plague deep learning-based intelligent technologies around us. APIs could be used to steal ML models. Unexpected instructions could be carried out by intelligent speech systems. Real-world image classifiers could be fooled by 3D-printed items [3]. Furthermore, safety-critical technologies need extensive security testing before they can be widely deployed, in order to assure their safety. Many key scholars and practitioners who interested in the security of deep learning in recent years. are looking into and researching possible assaults on deep learning systems, as well as protection mechanisms [3].

ML solutions have been effectively implemented in a variety of situations, but their use in the cyber security field is complicated and yet in its infancy. Researchers focus on adversarial attacks that aim to alter the detection and prediction capabilities of ML models, which is one of several central topics that concern security systems based on ML [5]. We look at real-world poisoning and evasion attacks aimed against malware, spam, and network intrusion-detection protection systems. In the context of IDSs, we look at the potential harm that an attacker could do to a cyber-detector and offer several known and novel defensive approaches. Several performance assessments are included in the study, all of which are based on long experimentation with big traffic datasets [5]. ICSs can monitor processes remotely and disseminate information using the Internet's ubiquitous interfaces, application, and infrastructure components. Many technology advancements offer new construction options for traditional ICSs (e.g., embedded, multi-standard network, and wireless technology). ICSs are exposed to a wide spectrum of aggressive cyberattacks as they shift from segregated to different areas, despite the potential advantages of modern information and communications technologies. [13]. There could be serious consequences for public safety and the economy should an ICS be disrupted. As a result, developing effective tools to detect malicious attacks against ICSs is crucial.

In image processing, adversarial assaults against ML have been investigated, but appropriate evaluations in the cybersecurity arena are lacking. A small number of cybersecurity challenges, a small number of ML classifiers, and a small fraction of adversarial assaults are considered in the publications that analyze the effectiveness of cyber detectors in adversarial scenarios. The major emphasis is on intrusion analysis; however, this study explores at this from the standpoint of network intrusion detection. We provide a complete review of adversarial assaults against ML- based cybersecurity solutions and offer a threat taxonomy in three categories: network intrusion detection. We review known strategies to combat these threats and offer a novel strategy for mitigating data poisoning assaults. We further conduct a large number of tests to evaluate and compare the performance of cyber detectors in both normal and hostile environments, as well as to assess the efficacy of several countermeasures, including the technique suggested in this study.

Previous studies are just conducting an adversarial attack in the ICS environment without a design a security solution or a trained model to detect the adversarial attacks [15]. Therefore, our main contribution is listed below:

- Proposing an IDS that can be used to detect malicious assaults on ICSs.
- Confirming that ICS security policies can be breached by malicious actions detected by an IDS.
- Highlighting the importance of system administrators using the information they gather from these assaults to take appropriate action [14].
- Highlighting that the future of ICS security must rely heavily on the development of robust intrusion detection technology.

2. Literature review

This section discusses a literature review dedicated to the topic adversarial attacks and what have researchers, scholars and field experts reached on detecting and circumventing such attacks are presented. An increasing number of ICSs are being protected by ML based IDSs. An overview of current ICSs and accompanying supervised learning algorithms for attack detection and categorization in various situations in the literature. There has been little attention paid to AML in this setting until recently. Researchers have recently become interested in phishing mails classifications, malware scanning, and AML against networking IDSs for network management [28]. Furthermore, both Nelson et al. and Zhou et al. show that simply modifying a small fraction of the initial learning algorithm, an attacker can attack and effectively circumvent ML techniques used in junk mail [8]. Grosse et al. also tested a neural net built on the DREBIN Mobile malware database. They claim that a modest amount of feature perturbation in the training set can confound the model. To be effective, a white - box testing attack requires the opponent to have exposure to or awareness of the information and the characteristics it includes. It was also shown that 170K Android apps were tested throughout 2017 and 2018 to show that escaping state of the art malware classifications is possible. It acquired less than five minutes in total to create adversarial programs, suggesting that "intelligent spyware" is a major threat because thousands of genuine and unobtrusive antagonistic apps can be easily obtained at mass. Hu and Tan offer a more advanced aggressive technique that uses deep neural networks (DNNs) to activities aimed virus classifications without having any information from the data or platform [9]. This is referred to as a "black-box" attack, which is a sort of assault. Conclusively, Appruzzese, Colajanni, and Marchetti [9] launch actual adversary operations against networking IDSs with the goal of detecting malware activity using classifier model. In the study, such assaults were proven to be successful.

Merely a few investigations regarding AML assaults have been conducted in the context of ICSs. An AML attack on an LSTM classifier applied to an ICS dataset was demonstrated by Zizzo et al., [10] who demonstrated a basic AML attack on the classifier. Because work in this field is still in its early stages, the hostile sampling was created by hand-selecting the obtained features to be interrupted. Yaghoubi and Fainekos [11] used a Simulink water condensing method to assess a horizontal stripe search approach. However, this technique is very effective against a restricted variety of systems that use smoothing convolution layers in recurrent neural networks (RNNs). They also employed an auto - encoder to generate antagonistic sample to demonstrate two separate sorts of genuine escape attacks using RNN models. Neither of the above research go into great detail about AML defenses. Finally, contemporary guided learning-enabled IDSs in ICSs have room to investigate AML and the protection against such assaults [11]. In an ICS environment, intrusion detection methods like as naive Bayes, regression trees, SVMs, and J48 are actually more prevalent. As a result, the investigations are focused mainly on constructing AML defenses utilizing these approaches, which are at the forefront of ML-driven ICS warning systems [12].

To combat specific network invasions, Wang et al. [19] suggested an intrusion detection system. To overcome the long training time and poor detection accuracy of conventional neural network models, network intrusion detection uses the Stacked Denoising Auto Encoder-Extreme Learning Machine (SDAE- ELM). The DBN-Softmax-based network intrusion prevention model aims to increase the identification efficacy of host invasion. During training process and refinement, comparatively tiny slope decline is utilised to increase both the learning happens and principles are derived of the networks. Investigations were performed out on a variety of datasets to verify that this concept outperformed other traditional machine learning methodologies. SDAE is a deep training algorithm for compressing large datasets. However, the majority of the samples included in this study are obsolete, and the average classification model recognition rate still has to be improved. The amount of attacks on ICS, such as malware and Trojans, has been constantly rising since 2016.

Industrial automation failures induced by targeted attackers, such as the Stuxnet [23] malware attacking an Iranian nuclear power plant in 2010, alert the user for ICS cyberspace security flaws. ML-based intrusion detection systems (IDS) can

function well in internet traffic identification in ICS. To be much more precise, machine learning solutions enable sensing devices to uncover patterns in enormous amounts of historical information. Furthermore, by employing labor to set classification model for characteristics or combinations of characteristics, it reduces the high misdiagnosis alerts that are generated. As a result, in ICS, ML-based IDS can be used as a supplement to regulation IDS [23]. ML-based IDS can achieve excellent detection precision and minimal probability of false alarm by meticulously studying network information. Many advanced cyber-attacks [22] utilize specific network assault primitives to infiltrate from company networks into SCADA networks, identify the involvement of various SCADA hosts, and cause harm to complex processes, according to recently published cybersecurity threats.

Traditional attacks leveraging weaknesses in IT network protocols and specialized cyber-attacks on SCADA network protocols are the two types of primitive network-based hacking attempts. In investigations on ICS protection, such as classifications and evaluations of cyber-attacks on SCADA communication protocol can be discovered. A SCADA network intrusion detection system must consider both traditional and specialized threats to identify actual threats [29].

Both Nelson et al. [14] and Zhou et al. [15] show that by altering a tiny fraction of the underlying training examples, an attacker can attack and effectively circumvent the machine learning methods used in spam detection. Grosse et al. [16] also test the resilience of a neural network that was developed on the DREBIN Android malware dataset. They claim that troubling a tiny variety of features in the training phase can cause the algorithm to get confused. Such an assault is known as a white box attack because the opponent must have exposure to or awareness of the information and the characteristics it contains in order to succeed. Furthermore, during 2017 and 2018, Pierazzi et al. [17] examined 170K Android applications to show the corresponding of dodging state-of-the-art virus detectors. Their findings demonstrated that "adversarial-malware as a service" is a dangerous issue, as it was easy to build thousands of genuine and unobtrusive antagonistic apps at scale, with a median generation time of only a few minutes.

3. Research methodology

In this section, the methodology used in this study was explained as well as the research requirements required to complete this study was described below. In the 1990s, the Inter-Control Center Communications Protocol (ICCP) was devised and submitted to the Electro technical Council by the American Electric Power Research Institute (EPRI) (IEC) [20]. In the electricity sector, the ICCP is primarily used to connect across multiple control centers. This method allows a client to interact with multiple cloud computers and vice versa. A user and a server should establish a bidirectional database with predictable network access to ensure an accurate data transfer. When contrasted to Modbus as shown in figure 1, the controlling access bidirectional database, which is utilized by the two hosts to describe constant identities, variable kinds, and user access, provides some further security. However, such security solutions are still connected with some security hazards [20]. First and foremost, they are vulnerable to assaults like as theft and counterfeiting leading to a shortage of encrypting data and personal authentication protocols. The bilateral charts can be manipulated with since they're not concealed.



Figure 1. MODBUS Protocol [20].

3.1 System requirements

As a requirement for conducting the attack scenario in the ICS, the system requirements are:

- **Dataset:** a dataset to conduct our attack scenario on in ICS. Dataset will be used to train and test classifiers to detect DDoS and black box adversarial attacks.
- **Creating samples using a model:** This approach seeks to create training records using deep neural network (DNN) models. The samples generated are comparable to those in the target training data set.
- **Simulation tool:** It will be used in order to simulate the attack scenario and the detection model.
- **Required computer specifications to run the simulation environment** are CPU with Intel(R) Core(TM) i5-10310U CPU @ 1.70GHz 2.21 GHz, RAM with 16.0 GB and Hard Disk with 500GB.

3.2 Datasets

The dataset [25] contains examples of both normal activities, as well as different attacks. The normal scenarios are controlled by an Auto IT Script. The attack scenarios are randomly chosen, and most originated attack and detection will be done using python model. The focus in our study will be on Normal, DDoS and Reconnaissance attacks as shown in Table 1 below.

Table 1. Dataset Attack Types [25]

Type of Attacks	Abbreviation
Normal	Normal(0)
Naïve Malicious Response Injection	NMRI(1)
Complex Malicious Response Injection	CMRI(2)
Malicious State Command Injection	MSCI(3)
Malicious Parameter Command Injection	MPCI(4)
Malicious Function Code Injection	MFCI(5)
Distributed Denial of Service	DDOS(6)
Reconnaissance	Recon(7)

The main issue with the gas pipeline datasets is that they are unsuited for IDS research in their current state. Every time there was a link between a parameter and an attack, the link might have been avoided. We employed a gas pipeline dataset in our research and used ten functions, which are listed below.

- **Gas Pipeline Dataset**
 - o **command_address**
 - If the command address isn't 4, Visual Studio Code/Google Colab considers it as a DOS assault. Some of this is beneficial, because the device transmitting commands has a MODBUS address of 4, so anything else might easily be an attacker. A few man-in-the-middle attacks posing as device 4 would, however, add some randomness.

- o **response_address**
 - This value is always 4 or 0 when it's a recon attack, and only 0 when it's a recon attack. This is due to the fact that 0 is the MODBUS address of a broadcast message, and reconnaissance attacks use broadcast messages to obtain the address of a responding device. Because broadcast messages are uncommon in an established ICS system, some of this is genuine. Broadcast messages, on the other hand, are a valid MODBUS function, therefore adding proper broadcast messages might assist to increase randomization.
 - o **response_length**
 - Unless it's a reconnaissance attack, this is always 19. After that, it's 123. This is due to the fact that the device response to a broadcast message is 123 bytes long. As a result, adding valid broadcast messages to the mix would assist to give some variety.
 - o **comm_read_function**
 - Except in the case of a DOS attack, this value is almost always 3. Because 3 is the MODBUS read registers function code, this is the case. However, because there is no specific "read" field in the MODBUS data, only a function code, this option is unnecessary. To get just the MODBUS function code, combine this field with resp read fun and subfunction.
 - o **resp_read_fun**
 - This number is always three or one. CMRI occurs only when the temperature is 1. To get the MODBUS function code, combine this field with comm read function and subfunction.
 - o **subfunction**
 - There are just three subfunction values: 0, 1, and 4. Unless it's an MFCI attack, it's always 0. This parameter is also unnecessary. Because this field should be used in conjunction with comm read function and resp read fun to provide the MODBUS function code.
 - o **setpoint**
 - Setpoint has only four distinct values: 20, 70, 80, and 90. It's an MPCFI attack if the setpoint isn't 20. More randomness might be readily provided by legitimately altering the setpoint to a wide variety of pressures.
 - o **control_mode**
 - Control mode can only be one of three values: 0, 1, or 2. MSCFI is almost always number one. 1 is for the pipeline's manual mode. To introduce randomization, the system should be operated in all three modes (manual, automatic, and off).
 - o **control_scheme**
 - The only values for control mode are 0 and 1. It's an MSCFI attack if it's 0.
- The control mode parameter specifies whether the system is in "pump" or "solenoid" mode. To provide unpredictability, the system should be operated in both valid modes.
- o **measurement**

All of the CMRI attacks had the same measurement range, ranging from 6 to 11. To enhance randomization, the CMRI attacks should be spread out more. All of the NMRI attacks have a value of greater than 100 or less than -1. This is permissible for attacks such as negative sensor measurement, sensor measurement that is significantly out of boundaries, or random sensor measurement, because all of these assaults will result in anomalous measurements. In our research we selected the following parameters in order to simulate our attack and detect scenarios:

 - command_address
 - response_address
 - response_length
 - comm_read_funtion

Those parameters have been selected in our study because in our attack and detect scenarios we need to focus on the DDoS, DoS and Reconnaissance attacks and show how can our intrusion detection system can capture and analyze those attacks in the ICS environment.

3.3 Design architecture

This section provides an overview of the overall design architecture of the whole research that includes which are described below:

- Design Flowchart: a flowchart design of the process of this project as illustrated in Figure 2.
- An abstract overview of the ICS: a high-level overview of industrial controls systems as depicted in Figure 4.

- Deployment Architecture Model: This study ICS attack and detection deployment architecture model as shown in Figure 5.

The following is flowchart process life cycle for our design architecture shown below in Figure 2. This project starts with data collection phase in which we collected the required data. Then the collected data is processed in data processing phase, after we processed the collected data, we will create the model for adversarial attack on control systems, then we will conduct an attack on the gathered processed data, then we will test and validate the created model. Finally, it will result with an attack detection prediction and we will try it on gas plant update and will update the model and continue on validation and testing the proposed model.

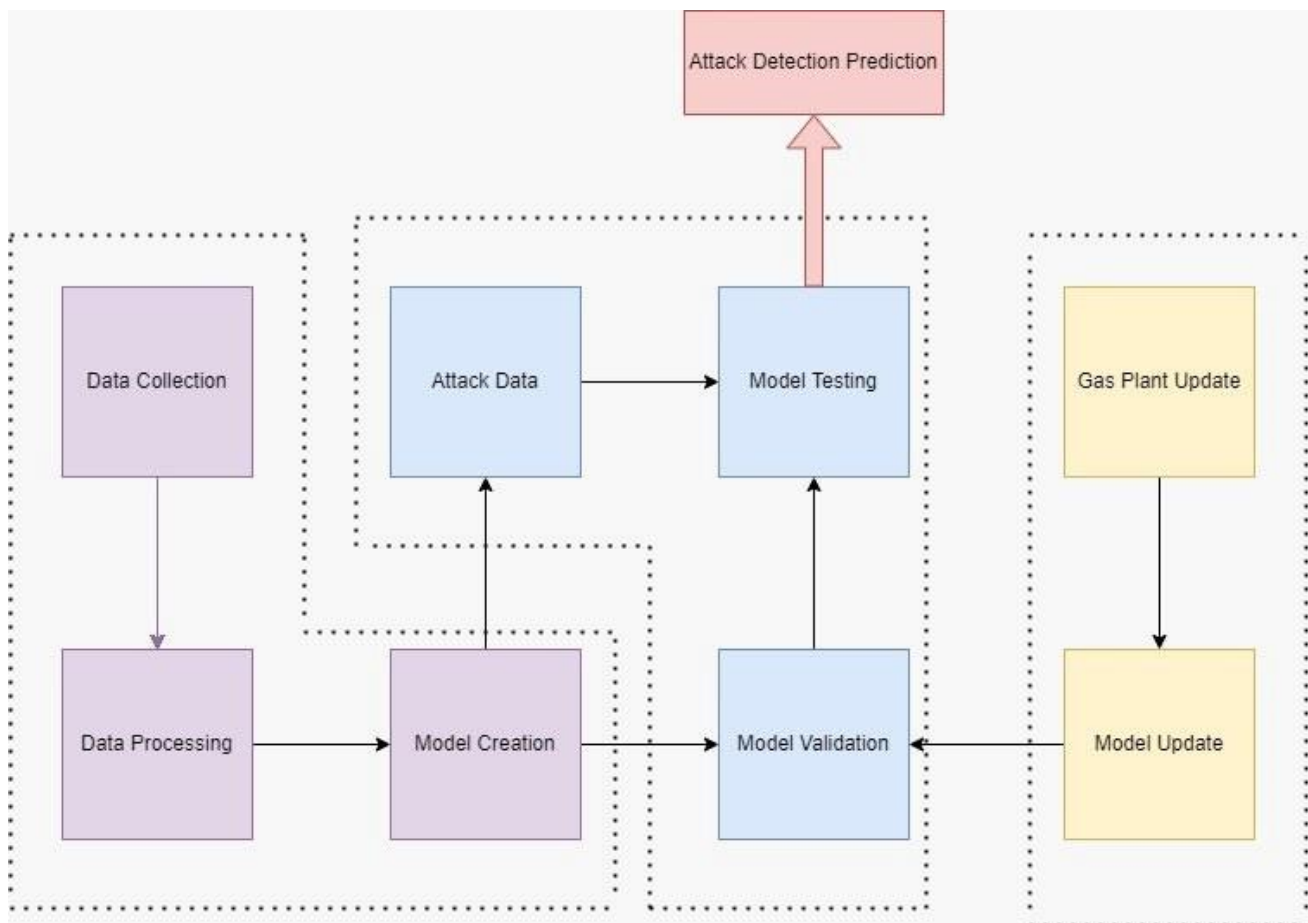


Figure 2. Design Flowchart.

Figure 3 depicts a gas plant pipeline in the two primary components, which are stated below:

(1) Human Machine Interface (HMI)

It can also display status information and historical data collected by the ICS devices. It's also used in controllers to monitor and configure set points, control algorithms, and alter and set parameters.

(2) Programmable logic controllers (PLCs)

It's directly connected to physical processes such as wastewater treatment plants, gas pipelines, and electrical power grids in industrial control systems (ICS). They have control logic built in that defines how to control and monitor the processes' activity.

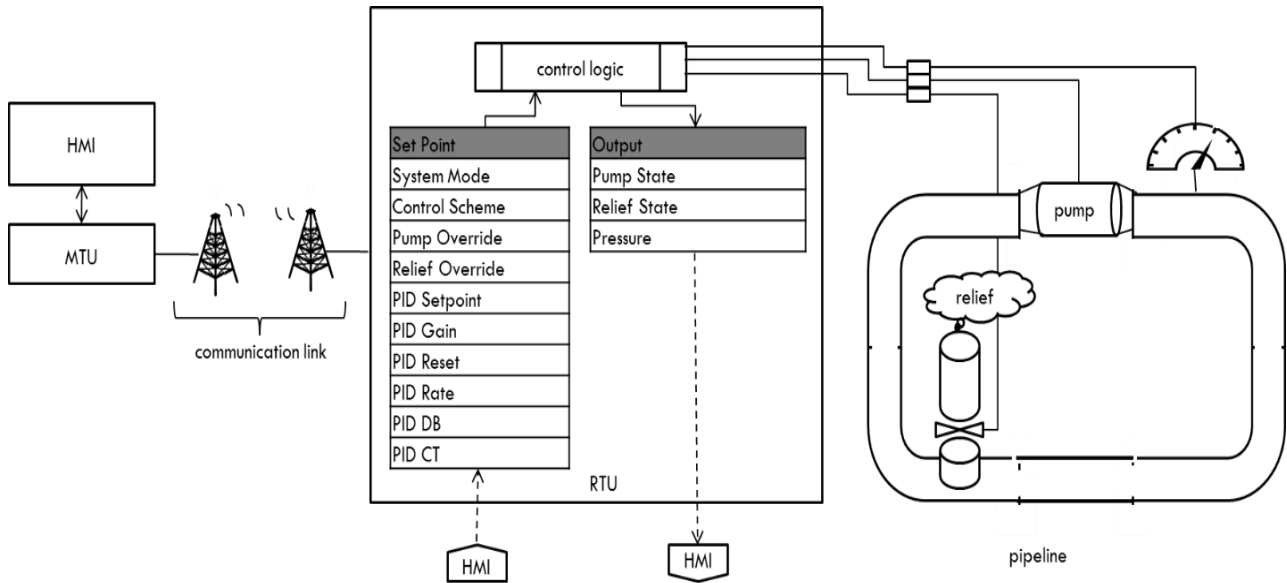


Figure 3. Gas Plant Pipeline [25]

Figure 4 is an abstract picture of an ICS. The physical layer's field devices, including as sensors and actuators, monitor and govern the underlying industrial process. Sensors sample the current condition of the process, which is then conveyed to the distributed PLCs. Pumps, valves, generators, and circuit breakers are all controlled by PLCs, which produce control actions and pass them to actuators. Other devices in the supervisory control layer, such as the SCADA and HMIs, provide communication between a plant operator and the PLCs for implementing human-assisted control operations. According to recent surveys, the vast majority of ICS systems employ proprietary communication protocols.

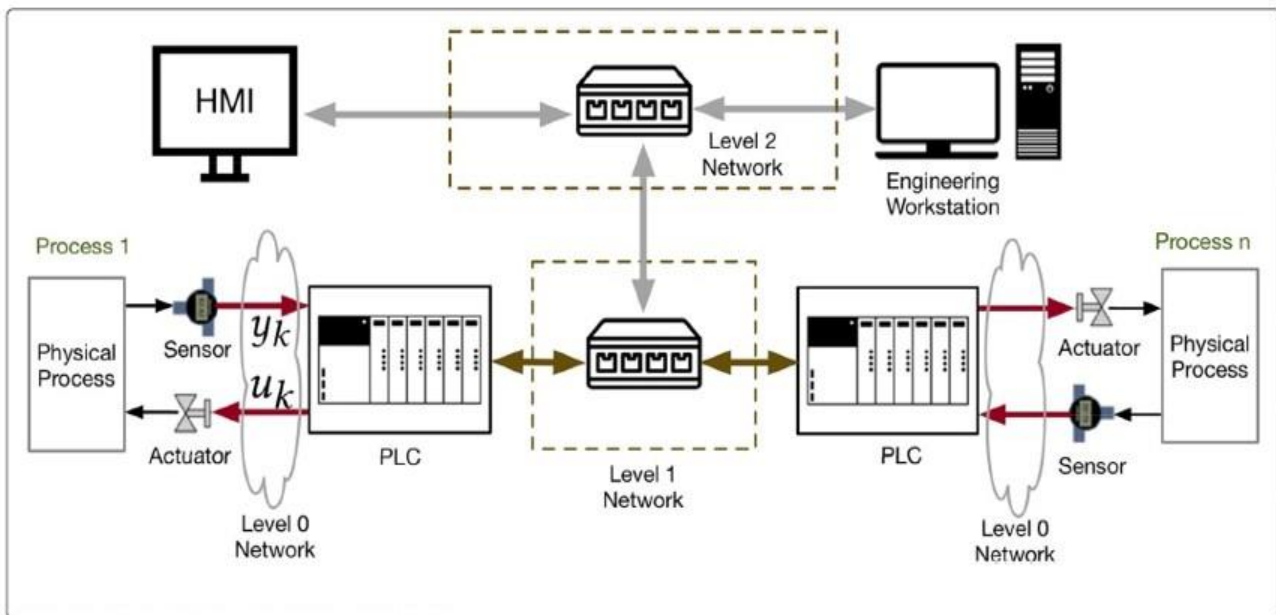


Figure 4. An abstract view of an ICS [24]

3.4 Deployment Architecture

The following is the study's deployment architecture which shown in Figure 5 we are placing the IDSs in 3 levels:

- Level 2 (Process Network)
- Level 3 (Operation ICT/DMZ)
- Level 4 (IT Network)

These three levels are needed to capture and analyze all the traffic received from the three levels to a get better understanding of our ICS environment and then we can keep monitoring and detecting all the traffic. In the process network we are placing the IDS before the HMI Local SCADA/DCS because we need to capture all the traffic that are placed before the control systems and in level 3 the IDS have been placed before the firewall so can capture all the traffic received before it will reach to the DNS and historian server. Finally, in the IT network we are placing the IDS before the corporate firewall for analyze all the traffic received before it reaches out to a remote access. In this deployment architecture we can have a centralized monitoring and detection of the attacks in all the 3 levels.

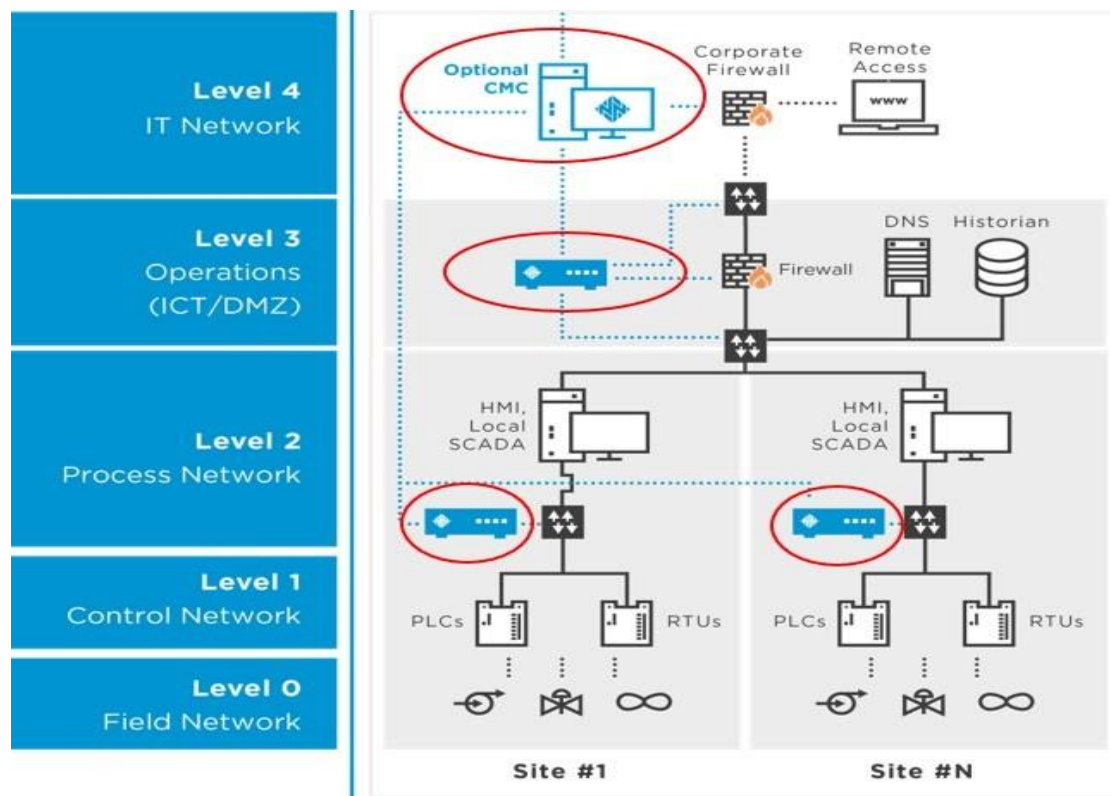


Figure 5. Deployment Architecture.

3.5 Approaches

Multiple methods for generating adversarial samples employ gradients, which are typically unavailable to an attacker in real-world circumstances. The adversary must generate adversarial perturbations without access to the target models to compute gradients in our approach, which we call black-box adversarial attacks. Previous methods attempted to estimate the gradient using either a transfer gradient from a surrogate white-box model or query feedback [30].

3.5.1 Attack Scenario

In this study we will conduct a DDoS, DoS and Reconnaissance attacks model trained by black box adversarial attack in the industrial control system and we are targeting a gas power plant that have pump will affect the availability on the ICS environment.

3.5.2 Association Rules

Association rules are generated using frequent item sets, as shown in equation 1 [25]. Associative rules are defined as rules that meet the minimum confidence requirement:

(1) Confidence: The equation's rule can be broken down into two pieces. One is 'X,' which is on the left side of \Rightarrow and is known as the antecedent, and the other is 'Y,' which is on the right side of \Rightarrow and is known as the consequent. The combined support of antecedent and consequent, as well as the antecedent alone, are used to assess the confidence of any rule.

$$C(XX \Rightarrow Y) = \frac{S(XXUY)}{S(XX)} \quad (3)$$

(2) Algorithm: Attack Generation using Association Rule Mining. The following show the steps of the algorithm and each step is dependent on another step:

1. Data acquisition via network packet capture

In this step we need to capture and collect all the data using a network packet capture

2. Sensor and actuator status information is stored by decoding network packets.

In this step we need to decode all the network packet to store the state of the sensors and actuators information details

3. Send the historian the state information.

In this step we need to send all the information states to be send to the historian server that is placed in the ICS environment

4. Feature/ Attribute Transformation, then feature selection using the acquired data.

In this step we need to select the feature/attributes which we are targeting by collecting all the data and information

5. using the modified data, create frequent item sets.

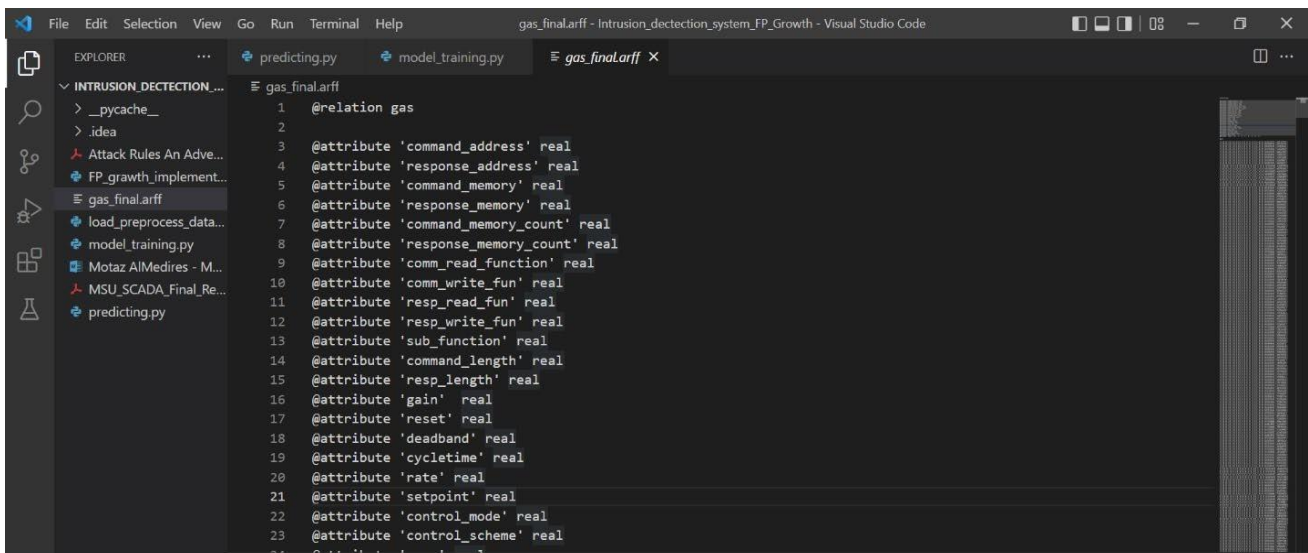
6. The frequent item sets are used to generate association rules.

7. Validation

In this step we need to validate the attacks (association rules) against the normal patterns

4. Implementation

The following section provides a sample of the overall implementation. First, we need to define the dataset attributes functions as you can see below our dataset are defined in the `gas_final.aff` an attribute-relation file format (AFF) file is an ASCII text file that describes a list of instances sharing a set of attributes.



```

1 @relation gas
2
3 @attribute 'command_address' real
4 @attribute 'response_address' real
5 @attribute 'command_memory' real
6 @attribute 'response_memory' real
7 @attribute 'command_memory_count' real
8 @attribute 'response_memory_count' real
9 @attribute 'comm_read_function' real
10 @attribute 'comm_write_fun' real
11 @attribute 'resp_read_fun' real
12 @attribute 'resp_write_fun' real
13 @attribute 'sub_function' real
14 @attribute 'command_length' real
15 @attribute 'resp_length' real
16 @attribute 'gain' real
17 @attribute 'reset' real
18 @attribute 'deadband' real
19 @attribute 'cycleTime' real
20 @attribute 'rate' real
21 @attribute 'setpoint' real
22 @attribute 'control_mode' real
23 @attribute 'control_scheme' real

```

Then we have four main python tabs that are used to load and preprocess our datasets, reduce the dataset into a readable format and import the association rules, trained our model and finally to simulate our attack and detection scenarios.

- Load_preprocess.py

In this class we need to import the most important libraries including Pandas, Arff, Seaborn and Matplotlib.pyplot.

```
• import pandas as pd  
• from scipy.io import arff  
• import seaborn as sns  
• import matplotlib.pyplot as plt
```

Then, we need to define our dataset file location path and print them into a set of columns.

```
•  
• data = arff.loadarff(r'C:\Users\malmedires001\Documents\Documents -  
28_Mar_21\Windos - Laptop\Projects\CODE\Final  
Code\Intrusion_dectection_system_FP_Growth\gas_final.arff')  
• df = pd.DataFrame(data[0])  
• print(df.head())  
•  
• # print(df.dtypes)  
•  
• # print(df.columns)
```

Then we need to print our reduce dataset in order to have it in a readable format.

- `for col in reduced_dataset.columns:`
- `print(reduced_dataset[col].value_counts())`
-
- `reduced_dataset['command_address'] = [0 if x == 4 else 1 for x in reduced_dataset['command_address']]`
- `reduced_dataset['response_address'] = [0 if x == 4 else 1 for x in reduced_dataset['response_address']]`
- `reduced_dataset['resp_length'] = [0 if x == 19 else 1 for x in reduced_dataset['resp_length']]`
- `reduced_dataset['comm_read_function'] = [0 if x == 3 else 1 for x in reduced_dataset['comm_read_function']]`
- `reduced_dataset['resp_read_fun'] = [0 if x == 3 else 1 for x in reduced_dataset['resp_read_fun']]`
- `reduced_dataset['sub_function'] = [0 if x == 0 else 1 for x in reduced_dataset['sub_function']]`
- `reduced_dataset['setpoint'] = [0 if x == 20 else 1 for x in reduced_dataset['setpoint']]`
- `reduced_dataset['control_mode'] = [1 if x == 1 else 0 for x in reduced_dataset['control_mode']]`
- `reduced_dataset['control_scheme'] = [0 if x == 1 else 1 for x in reduced_dataset['control_scheme']]`
- `reduced_dataset['measurement'] = [1 if (x>=6 and x<=11 or x>=100) or x<=-1 else 0 fo`

```
•  
• preprocessed_binary_data = reduced_dataset  
•  
• print("\n Binary converted dataset\n")  
• for col in preprocessed_binary_data.columns:  
•     print(reduced_dataset[col].value_counts())
```

FP_growth_implementation.py

In this class we are importing and loading our reduced dataset

```
• from load_preprocess_data import reduced_dataset  
• from mlxtend.frequent_patterns import association_rules  
• import numpy as np  
• import pandas as pd
```

5. Results and Discussions

In this study we conducted an adversarial attack and detection on industrial control system to protect the availability of the ICS and be able to detect such attacks. We have used a dataset as explained in section 5.1 that is used for gas pipeline plant as discussed in section 5.2. Furthermore, we trained adversarial model in ICS and conducted three types of attacks: DDoS, DoS, Reconnaissance on all three levels of ICS levels of architecture as detailed in section 5.3. A deep neural network DNN algorithm used in our study as explained . Tools used in order to achieve the results of this study are visual studio, google colab, python programming language and is further discussed below. The following subsections are the three attack types simulated in this study and its results.

5.1 Distributed Denial of Service Attack (DDoS) Results:

In this experiment we simulated a distributed denial of service attack on a trained dataset on a ratio of 0.20 for testing and 0.80 for training. `command_address` and `comm_read` functions utilized to establish and detect the DDoS attack. Results collected from this experiment is that we were able to successfully train an adversarial attack through DDoS. Details of this experiment is further explained in Appendix A. Furthermore, we were able to successfully detect a DDoS attack through adversarial attack model. Figure 6 illustrates the results gathered from this experiment.

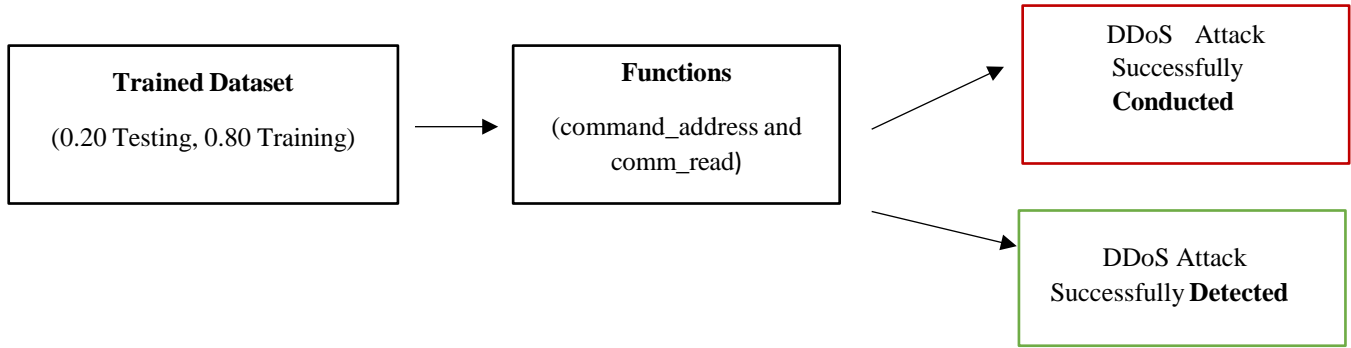


Figure 6. Distributed Denial of Service Attack (DDoS) Results

5.2 Denial of Service Attack (DOS) Results

In this experiment we simulated a distributed denial of service attack on a trained dataset on a ratio of 0.20 for testing and 0.80 for training. `command_address` and `comm_read` function utilized to establish and detect the DoS attack. Results collected from this experiment is that we were able to successfully train an adversarial attack through DoS. Furthermore, we were able to successfully detect a DoS attack through adversarial attack model. Figure 7 illustrates the results gathered from this experiment.

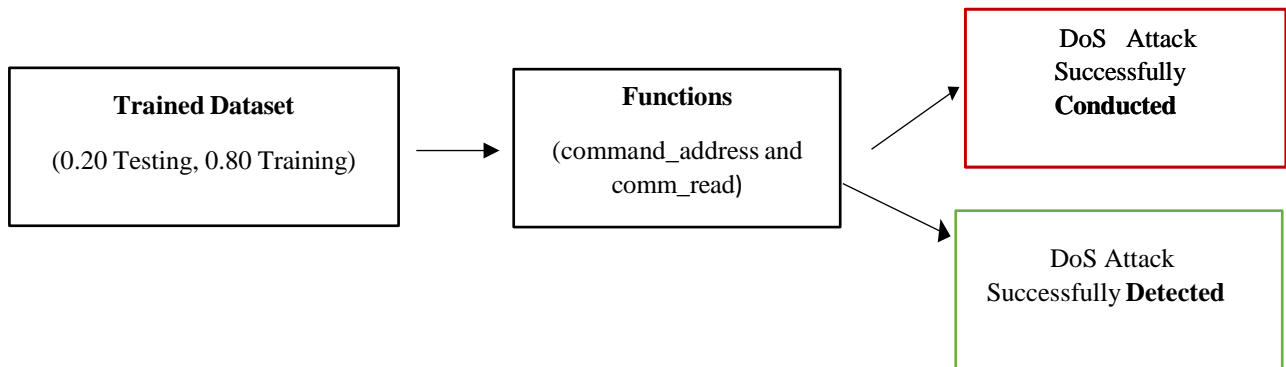


Figure 7. Denial of Service Attack (DoS) Results.

5.3 Reconnaissance Results

In this experiment we simulated a reconnaissance attack on a trained dataset on a ratio of 0.20 for testing and 0.80 for training. `response_length` and `response_address` function utilized to establish and detect the reconnaissance attack. Results collected from this experiment is that we were able to successfully train an adversarial attack through reconnaissance. Details of this experiment is further explained in Appendix A. Furthermore, we were able to successfully detect a reconnaissance attack through adversarial attack model. Figure 8 illustrates the results gathered from this experiment.

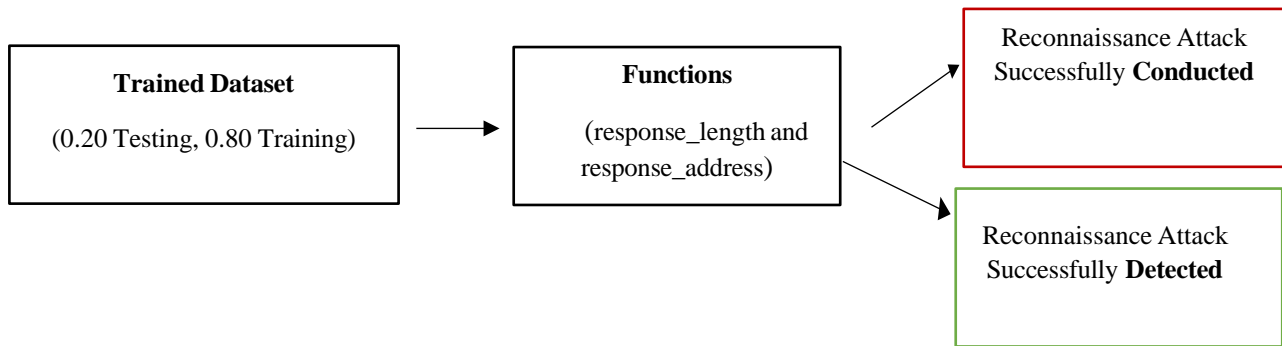


Figure 8. Reconnaissance Attack Results.

In this study, we conducted three types of attacks DDoS, DoS and Reconnaissance on an ICS environment simulated surface in an adversarial attack architecture model. The result of this study is that we were able to conduct these types of attacks and detect them. This study will strengthen the availability factor of industry control systems in order to have an available efficient performance of an industry control systems.

6. Conclusion

In this study, we conducted three types of attacks DDoS, DoS and Reconnaissance on an ICS environment simulated surface in an adversarial attack architecture model. The result of this study is that we were able to conduct these types of attacks and detect them. This study will strengthen the availability factor of industry control systems in order to have an available efficient performance of an industry control systems and proposing an IDS that can be used to detect malicious assaults on ICSs. Finally confirming that ICS security policies can be breached by malicious actions detected by an IDS. We also need to keep into consideration that deep learning has carved itself a place in the technological arena as a result of recent breakthroughs in the field of AI, and it is now being employed in autonomous and IoT systems all over the world. Unfortunately, adversarial assaults on deep learning models have grown common, posing a serious threat to their integrity. Many state-of-the-art models have been found to be vulnerable to assaults by well-crafted adversarial instances. These adversarial examples are tainted copies of clean data with a modest bit of noise. These hostile samples are undetectable to the naked eye; however, they are quite effective at fooling the targeted model. The vulnerability of these models raises concerns about their suitability for safety-critical real-world applications, such as autonomous driving and medical applications. Huang et al. (2017) revealed that the intriguing attack mode adversarial assault is equally successful when attacking neural networks under reinforcement learning, sparking new study in this area. Our study examines related contributions, focusing on the most significant and illuminating works on the subject. We provide a detailed overview of the literature on adversarial assaults in various reinforcement learning applications, as well as a brief analysis of the most effective mitigation methods against existing adversarial attacks. ML-based IDSs are now recognized as essential tools for detecting cyberattacks in ICSs due to their effectiveness and versatility.

As a result, these systems are vulnerable to AML assaults, which can severely impair or mislead their capabilities. Adversaries could potentially change data points maliciously to avoid detection by the IDS, which would delay attack detection and result in considerable damage when launched against ICS infrastructures. Therefore, developing more powerful ML-based IDS requires understanding the applicability of these threats in ICSs. By producing hostile samples and examining categorization behaviors, adversarial training can also be used to attack confirmed the effectiveness. To complement the studies described above, a genuine energy infrastructure database was used to train and assess frequently used based on supervised models.

The attacker model and assumptions used in this work are also realistic. And were used to generate adversarial samples with various combinations of noise and feature perturbation, depending on the quantity of noise and how many features were perturbed. Accordingly, random forest and J48 approaches were used to evaluate these samples. In addition, the study investigated how adversarial training on such samples can promote the robustness of supervised models. A tenth of the adversarial data points created at random were even included in the initial training dataset. New adversarial samples were used to retrain and apply the models. Overall, the random forest model outperformed the J48 model when it came to

predicting JSMA parameters. According to the results of this study, the random forest model is the most effective model for categorizing adversarial samples in the given dataset.

7. Future works

Confrontational sampling can be produced with JSMA and have an effect on the classifier performance of existing model evaluation; however, there are various different approaches for creating such data (e.g., iterative gradient sign, Carlini Wagner, GANs). In the future, this research could be expanded to include other systems as a supply of antagonistic data. AML should also be contrasted with other algorithms, such as LSTMs. Finally, antagonistic learning was employed to demonstrate the controlled systems' resilience. To be honest, this method may not always be adequate since it is hard to forecast all forms of AML assaults that could be directed against a given system. As a result, further defense systems must be investigated. JSMA can be used to create antagonistic data, which have an influence on the classifier performance of existing classification purpose. It would be possible to extend this study to incorporate other models as a source of hostile samples in the future. AML should be compared to other models, such as LSTMs, for example.

Corresponding author

Motaz Abdulaziz Almedires
221445316.student@kfu.edu.sa

Acknowledgements

Not applicable.

Funding

No funding.

Contributions

M.A.A; Conceptualization, A.A; Investigation, A.E; Writing (Original Draft), M.A.A; A.A; and A.E Writing (Review and Editing) Supervision, M.A.A; A.A; and A.E Project Administration.

Ethics declarations

This article does not contain any studies with human participants or animals performed by any of the authors.

Consent for publication

Not applicable.

Competing interests

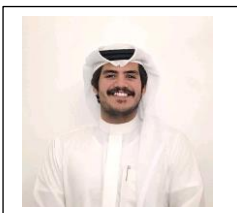
The author declares no competing interests.

References

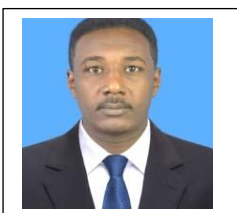
- [1] M. Krotofil and D. Gollmann, "Industrial control systems security: What is happening?" in 2013 11th IEEE International Conference on Industrial Informatics (INDIN), 2013, pp. 670-675.
- [2] E. Estévez and M. Marcos, "Model-based validation of industrial control systems," IEEE Transactions on Industrial Informatics, vol. 8, pp. 302-310, 2011.
- [3] M. Kravchik and A. Shabtai, "Detecting cyberattacks in industrial control systems using convolutional neural networks," in Proceedings of the 2018 Workshop on Cyber-Physical Systems Security and PrivaCy, 2018, pp. 72-83.
- [4] O. Andreeva, S. Gordeychik, G. Gritsai, O. Kochetova, E. Potseluevskaya, S. I. Sidorov, et al., "Industrial control systems vulnerabilities statistics," Kaspersky Lab, Report, 2016.
- [5] M.-K. Yoon and G. F. Ciocarlie, "Communication pattern monitoring: Improving the utility of anomaly detection for industrial control systems," in NDSS Workshop on Security of Emerging Networking Technologies, 2014.
- [6] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets and challenges," Cybersecurity, vol. 2, pp. 1-22, 2019.
- [7] V. Jyothsna, R. Prasad, and K. M. Prasad, "A review of anomaly based intrusion detection systems," International Journal of Computer Applications, vol. 28, pp. 26-35, 2011.
- [8] I. Butun, S. D. Morgera, and R. Sankar, "A survey of intrusion detection systems in wireless sensor networks," IEEE Communications Surveys & tutorials, vol. 16, pp. 266- 282, 2013.
- [9] T. H. Morris and W. Gao, "Industrial control system cyberattacks," in 1st International Symposium for ICS & SCADA Cyber Security Research 2013 (ICS-CSR 2013) 1, 2013, pp. 22-29.

- [10] H. Holm, M. Karresand, A. Vidström, and E. Westring, "A survey of industrial control system testbeds," in Nordic Conference on Secure IT Systems, 2015, pp. 11-26.
- [11] S. Ponomarev and T. Atkison, "Industrial control system network intrusion detection by telemetry analysis," IEEE Transactions on Dependable and Secure Computing, vol. 13, pp. 252-260, 2015.
- [12] E. Monmasson, L. Idkhajine, M. N. Cirstea, I. Bahri, A. Tisan, and M. W. Naouar, "FPGAs in industrial control applications," IEEE Transactions on Industrial Informatics, vol. 7, pp. 224-243, 2011.
- [13] T. H. Morris, Z. Thornton, and I. Turnipseed, "Industrial control system simulation and data logging for intrusion detection system research," 7th Annual Southeastern Cyber Security Summit, pp. 3-4, 2015.
- [14] H. R. Ghaeini and N. O. Tippenhauer, "Hamids: Hierarchical monitoring intrusion detection system for industrial control systems," in Proceedings of the 2nd ACM Workshop on Cyber-Physical Systems Security and Privacy, 2016, pp. 103-111.
- [15] Y. Hu, A. Yang, H. Li, Y. Sun, and L. Sun, "A survey of intrusion detection on industrial control systems," International Journal of Distributed Sensor Networks, vol. 14, p. 1550147718794615, 2018.
- [16] M. Caselli, E. Zambon, and F. Kargl, "Sequence-aware intrusion detection in industrial control systems," in Proceedings of the 1st ACM Workshop on Cyber-Physical System Security, 2015, pp. 13-24.
- [17] M. J. Assante and R. M. Lee, "The industrial control system cyber kill chain," SANS Institute InfoSec Reading Room, vol. 1, 2015.
- [18] M. Mantere, M. Sailio, and S. Noponen, "Network traffic features for anomaly detection in specific industrial control system network," Future Internet, vol. 5, pp. 460-473, 2013.
- [19] T. L. Blevins, "PID advances in industrial control," IFAC Proceedings Volumes, vol. 45, pp. 23-28, 2012.
- [20] O. Navarro, S. A. J. Balbastre, and S. Beyer, "Gathering intelligence through realistic industrial control system honeypots," in International Conference on Critical Information Infrastructures Security, 2018, pp. 143-153.
- [21] H. Abdo, M. Kaouk, J.-M. Flaus, and F. Masse, "A safety/security risk analysis approach of Industrial Control Systems: A cyber bowtie—combining new version of attack tree with bowtie analysis," Computers & Security, vol. 72, pp. 175-195, 2018.
- [22] Yang, H., Cheng, L., & Chuah, M. C. (2019a). Deep-Learning-Based Network Intrusion Detection for SCADA Systems. 2019 IEEE Conference on Communications and Network Security (CNS). <https://doi.org/10.1109/cns.2019.8802785>
- [23] Chen, J., Gao, X., Deng, R., He, Y., Fang, C., & Cheng, P. (2021). Generating Adversarial Examples against Machine Learning based Intrusion Detector in Industrial Control Systems. IEEE Transactions on Dependable and Secure Computing, 1–1. <https://doi.org/10.1109/tdsc.2020.3037500>
- [24] Anthi, E., Williams, L., Rhode, M., Burnap, P., & Wedgbury, A. (2021). Adversarial attacks on machine learning cybersecurity defences in Industrial Control Systems. Journal of Information Security and Applications, 58, 102717. <https://doi.org/10.1016/j.jisa.2020.102717>.
- [25] Morris, T.H., Thornton, Z. and Turnipseed, I., 2015. Industrial control system simulation and data logging for intrusion detection system research. 7th annual southeastern cyber security summit, pp.3-4.
- [26] Umer, Muhammad Azmi, et al. "Attack rules: an adversarial approach to generate attacks for Industrial Control Systems using machine learning." Proceedings of the 2th Workshop on CPS&IoT Security and Privacy. 2021.
- [27] Hsu, J., D. Mudd, and Z. Thornton. "Mississippi State University Project Report-SCADA Anomaly Detection." (2014).
- [28] Arora, Pallavi, Baljeet Kaur, and Marcio Andrey Teixeira. "Evaluation of machine learning algorithms used on attacks detection in industrial control systems." Journal of The Institution of Engineers (India): Series B 102.3 (2021): 605-616.
- [29] Cook, Allan, et al. "Attribution of cyber-attacks on industrial control systems." EAI Endorsed Transactions on Industrial Networks and Intelligent Systems 3.7 (2016).
- [30] Ren, Kui, et al. "Adversarial attacks and defenses in deep learning." Engineering 6.3 (2020): 346-360.

Biographies



Motaz Abdulaziz Almedires received a master degree in Cybersecurity from King Faisal University. He has an excellent experience in the cybersecurity field in both theoretical and practical. He has several certificates in cybersecurity like CEH and others. He several publications in cyber risk assessment. His research interests including cyber security, risk assessment and cyber-attacks. 221445316.student@kfu.edu.sa



Dr. Ahmed Elkhail is a distinguished Sudanese scholar with a robust computer engineering and technology background. I completed my B.S. in 2014 and M.Sc. in 2017 at the University of Gezira, Sudan, and earned my PhD from Southwest Jiaotong

University, China, in 2024. My research interests include cryptography, network security, blockchain technology, cloud computing, virtualization, and network architecture. I am an associate professor at the School of Computer and Information Engineering, Qilu Institute of Technology, China. engkhalil31@qlit.edu.cn



Dr. Mohammed Amin is an Associate Professor in the Department of Computer Science at University of Jordan. Almaayah is among the top 2% scientists in the world from 2020 up to now. He is working as Editor in Chief for the International Journal of Cybersecurity and Risk Assessment. He has published over 115 research papers in highly reputed journals such as the Engineering and Science Technology, an International Journal, Education and Information Technologies, IEEE Access and others. Most of his publications were indexed under the ISI Web of Science and Scopus. His current research interests include Cybersecurity, Cybersecurity-Risk Assessment and Blockchain.